

PDE

fields

#	string	fvt	fht	section
0	st storage type	0	F	a
1	nl name length	1	F	
2	nam name	2	7	
3	typ type	4	7	
4	kb key block	5	C	b
5	b <sub>1</sub> blocks used	6	E	
6	len length	7	9	
7	cre creation	8	B	
8	ver version --> minimum	9	10	c
9	acc access	A	9	
A	aux auxiliary	B	C	
B	mod last modification	C	10	
C	<del>hdb</del> header block	D	X F	d
D	res reserved	(b) 5	B	
E	enl entry length	B	X F	
F	epb entries per block	C	10	
10	fls file count	D	X D	e
11	bit bit map block	E	10	
12	bpu blocks per volume	F	X 10	
13	pbk parent block	E	F	
14	pen parent entry number	F	16	f
15	pel parent entry length	16	16	

idx # (message number)

idx fvt, x

sta vtab

jsr basr/c

txa

asl

tax

idx fieldtbl, x

an addr

ins

idx fieldtbl, x

scn address

ldy fteoff

jsr pstr

asto = E2

1028-1068



fs: a b c d  
vs: a e c f g  
ss: a e c f h  
etc → - -

fil fids.  
vol fids.  
sub fids.

to	1	2	3	4	5	6	7	8	9	A	B	C	D
fil fids.	0	1	2	3	4	5	6	7	8	9	A	B	C
vol fids.	0	1	2	3	D	7	8	9	E	F	10	11	12
sub fids.	0	1	2	3	D	7	8	9	E	F	10	13	14 15

etc (4)

PDE

dspfs\_ display field section

a

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19  
storage type: seedling  
1 name length: 5  
2 name: CHRIS e e e e e e e e e e e e e e e e  
3 type: f p f s e s

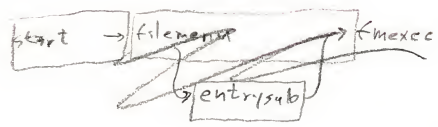


b

4 key block:  
5 blocks used.  
6 length:  
7 creation:

date:

dsr date 01/11/11 11:11:11



edit

```

    ldx curloff
    lda off11,x
    sta curaddr
    clc
    lda curpg
    adc off1b,x
    sta curaddr+1
  
```

```

    ecart → lda z curi
    bne
    dbv #0
    lda (curaddr),y
    and #%.ffff0000
    cmp #%.11110000
    bne
    sub!
    (fil)
  
```

exit

i (curaddr) → star top/ham len

if not at dir.header entry, then edit fil fids

; it/b1

; sz

; not dir bdy

# PLE

\* = data additional to hex

fields	VTAB	FVS	
st * storage type	0 0 0	0	X
nl name length	1 1 1	1	tspr
name * name	2 2 2	2	A
type * type	4 4 4	3	
kb key block	5	4	
bu blocks used	6	5	B
len length	7	6	
cre * creation	8 8 8	7	
ver version -> minimum	9 9 9	9	C
acc * access	A A A	9	
aur aur library	B	A	D
mod * last modification	E	B	
hdr header block	D	C	

\* str : address of string start  
(term. with 0)

months	x
Jan	0
Feb	1
Mar	2
Apr	3
May	4
Jun	5
Jul	6
Aug	7
Sep	8
Oct	9
Nov	A
Dec	B

months	address
Jan	0
Feb	1
Mar	2
Apr	3
May	4
Jun	5
Jul	6
Aug	7
Sep	8
Oct	9
Nov	A
Dec	B

f. elbcb

fvs	bits	res	reserved	D
bb	E	el	entry length	E
bb	F	eb	entries per block	F
cc		fc	file count	10
dd		fm	bit map block	11
ee		fv	blocks in volume	12
ff		pb	parent block	13
		pcn	parent entry number	14
		pel	parent entry length	15

code:  
lda #x  
asl  
tax  
lda frrtbl, x  
sta addr  
lax frrtbl, x  
sta addr  
ldy #tbl  
lsv prstr

display  
Section  
subroutines

file	0	1	2	3	4	5	6	7	8	9	A	B	C	D
vol	0	1	2	3	4	5	6	7	8	9	A	B	C	D
sub	0	1	2	3	4	5	6	7	8	9	A	B	C	D
fvol	0	1	2	3	4	5	6	7	8	9	A	B	C	D
vut	0	1	2	3	4	5	6	7	8	9	A	B	C	D
svt	0	1	2	3	4	5	6	7	8	9	A	B	C	D

typentbl  
typstbl

lda filetype  
sta asto  
lda #0  
lda typentbl, x  
lax  
cmp asto  
bne  
dex  
stx asto  
tax  
asl  
clc  
adc asto  
tax  
lda typstbl, x  
lax  
jsr cont  
tax  
lda typstbl, x  
lax  
jsr cont  
bts

storage type

sttbl	
0	deleted
1	seedling
2	sapling
3	tree
4	
5	
D	subdirectory
E	header (subdirectory)
F	header (volume directory)

9 5 b3  
9 3 2 7  
7 5 4



PDE

storage type: sapling  
name length: \$8  
name: CALCULTEVOX0000  
.....

type: sys

key block: \$xxxx

blocks used: \$xxxx

length: \$xxxx

208  
\$ 07 34 58 23 21 35 27 ...

4-2 *Borland*

FF

\$8 \$b

\$E \$b

b7 \$5 \$b

byte	field	bytes
2	name length	1
b	key block	2
b	blocks used	2
b	length	3
c	version → unknown	2
d	aux type	2
d	header block	2
e	reserved	8
f	entry length	1
f	entries per block	1
f	file number	2
f	bit map block	2
b	blocks in volume	2
h	parent block	2
h	parent entry number	1
h	parent entry length	1

byte	field	bytes
2	name	15
2	storage type	1
2	type	1

byte	field	bytes
4	date created	4
4	date modified	4
1	address	1

sub- / super-directories

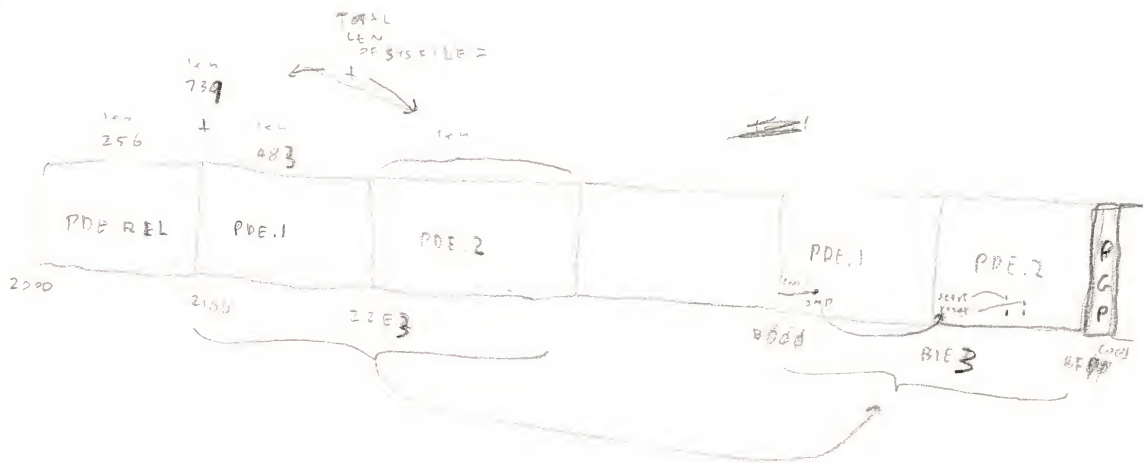
phf

2020/05/05

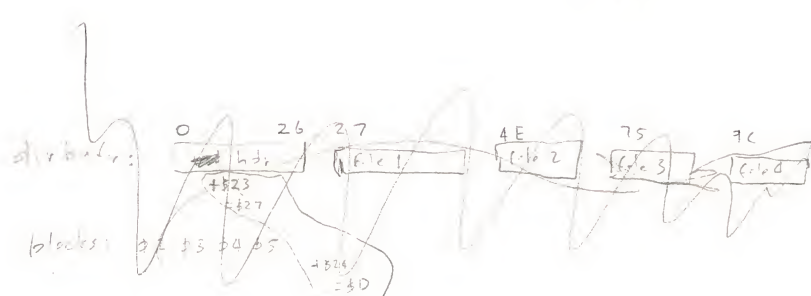
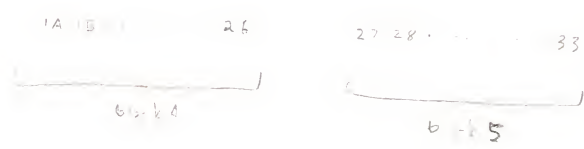
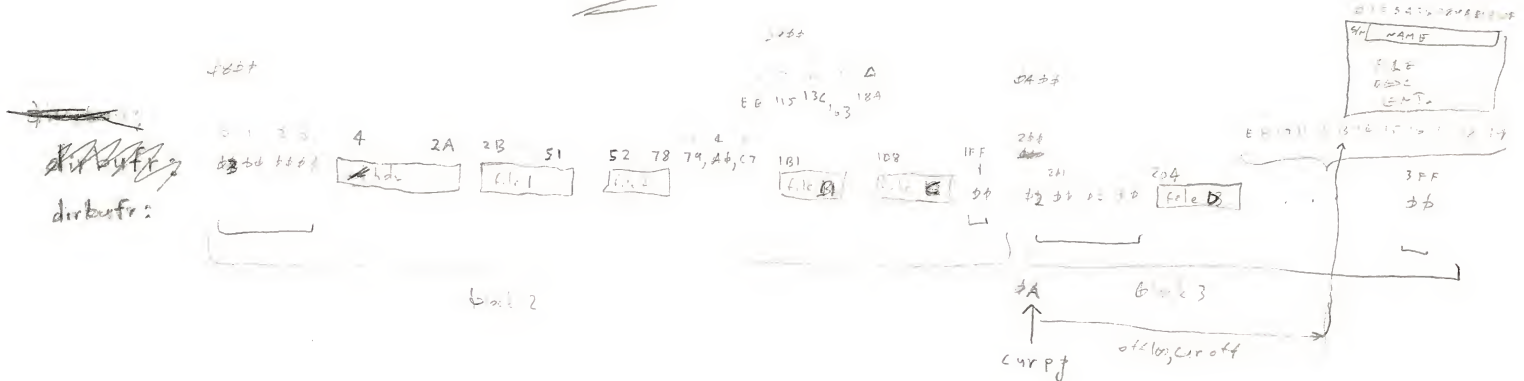
14 PDF.V

exit	Bφ3E	vtgφ	Eφ
firmess	Bφ4A	inmark	E1
cinout	Bφ6D	aslo	E2
cont	Bφ7B	bas	Fφ
zinout	BφAφ		
zout	BφAE		
nibout	BφB8		
nout	BφDA		
basalc	BφF3		
etch	BφφD		
waitkey	B11B		
dspetle	B127		
home	B172		
tlhome	B178		
cdown	B17C		
title	B1BA		
ctant	B1E2		
pcet	B1E5		

1038; 1058



PFE

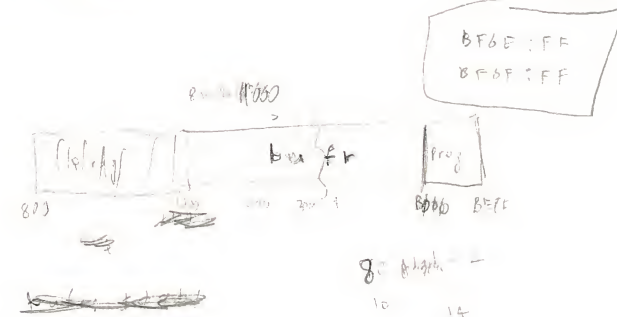


520 blocks  
(entries per block \* blocks) = 527  
entries per block = 50

idx of -1  
idx #0 f11  
set2 addr  
idx = curpf  
ade of 11h  
set2 addr

10 blocks maximum  
total = 527

### BINARY ALLOCATION



entry 8: 08

84 blocks max  
154 blocks max

5. step 12a machid  
and = 502  
low 5000  
imp exit  
5. step 12a 502  
imp entry ← 50300

exit 12a 515  
jsr entry  
jsr m1  
dlb quit  
dlb <ex1> evl  
exl dlb 504  
dlb 5, 6, 7, 8, 9

off11	04	2B	52	79	A7	C7	EC	15	3C	63	8A	B1	D8
off1h	DP	07	07	07	07	07	07	07	07	07	07	07	07
	1	1	1	1	1	1	1	1	1	1	1	1	1
	0	1	2	3	4	5	6	7	8	9	A	B	C

Hand-drawn scribbles and notes.

# POE test

B003	curi	C
4	curp	E
5	marp	E
6	cur off	C
7	invt	6: 11
8	topn	7

B1C6: followup

B3B: prevus

3	4	5	6
0	1	2	3

B4A: diving

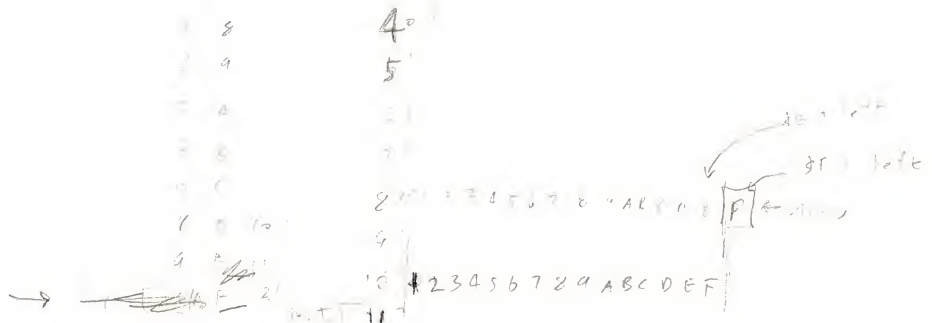
top	bot	11
0	1	2

B4A: off

B1C6: followup

B1F6: dspfwih

B275: dspflin



$$x = 0 \quad (cur off)$$

$$y = 8 \quad (cur off)$$

PDE

5

63

max

63

to 4 2  
div 4, 60 100 left  
4 14

top  
FILE 1  
FILE 2  
FILE 3  
FILE 4  
FILE 5  
FILE 6  
FILE 7  
FILE 8  
FILE 9  
bottom 10

4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
input

input bottom

curr =

(value of input)

input 1 + 1 = 2  
input 2 + 1 = 3  
input 3 + 1 = 4  
input 4 + 1 = 5  
input 5 + 1 = 6  
input 6 + 1 = 7  
input 7 + 1 = 8  
input 8 + 1 = 9  
input 9 + 1 = 10  
input 10 + 1 = 11

(minimum value for normal printing)

(= 5)

top = input - top  
(number of log above input)

(= 5)

for curr < top:

curr = curr - 1

print blank line

print FILE 1

print FILE (last) (vtab = tot)

after last file printed,  
print blank line (unless  
part vtab 23)

if  $\leq$  start input then  
cannot move down further

else other way: vtab top  
print FILE  
curr = curr - top  
curr = curr - top

# FSMON

title

equates

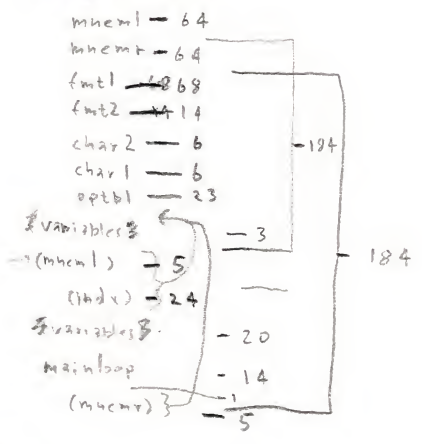
- opcodes
- page \$66 locations
- ~~page \$66 locations~~
- page \$66 locations
- ram locations



global table

- org
- jmp table (interface)
- cmd table
- jmp table (cmds)

disassembly table



- init
- copy
- page
- cbtoads
- cbtoads
- prline
- memset
- displace
- dsrfs
- dsprfs
- prins
- chout
- curorf
- scrollup
- scrollw

- prmess
- cinout
- cout
- cinout
- cout
- bascalc
- fetch
- execute

library routines

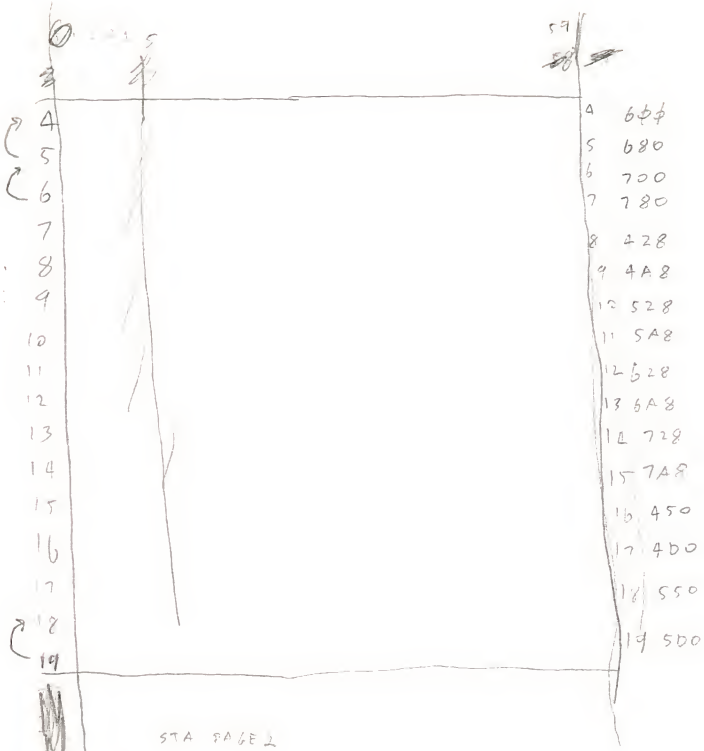
- hex
- up
- down
- left
- right
- begin
- end
- rtog
- mtog
- exit

coming and



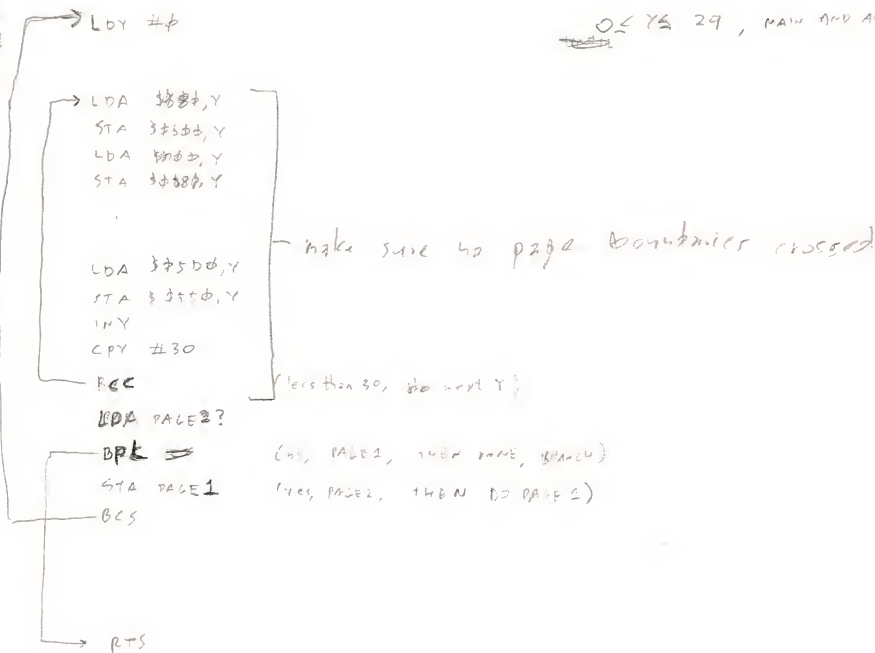
BLOXAP

SCROLL UP

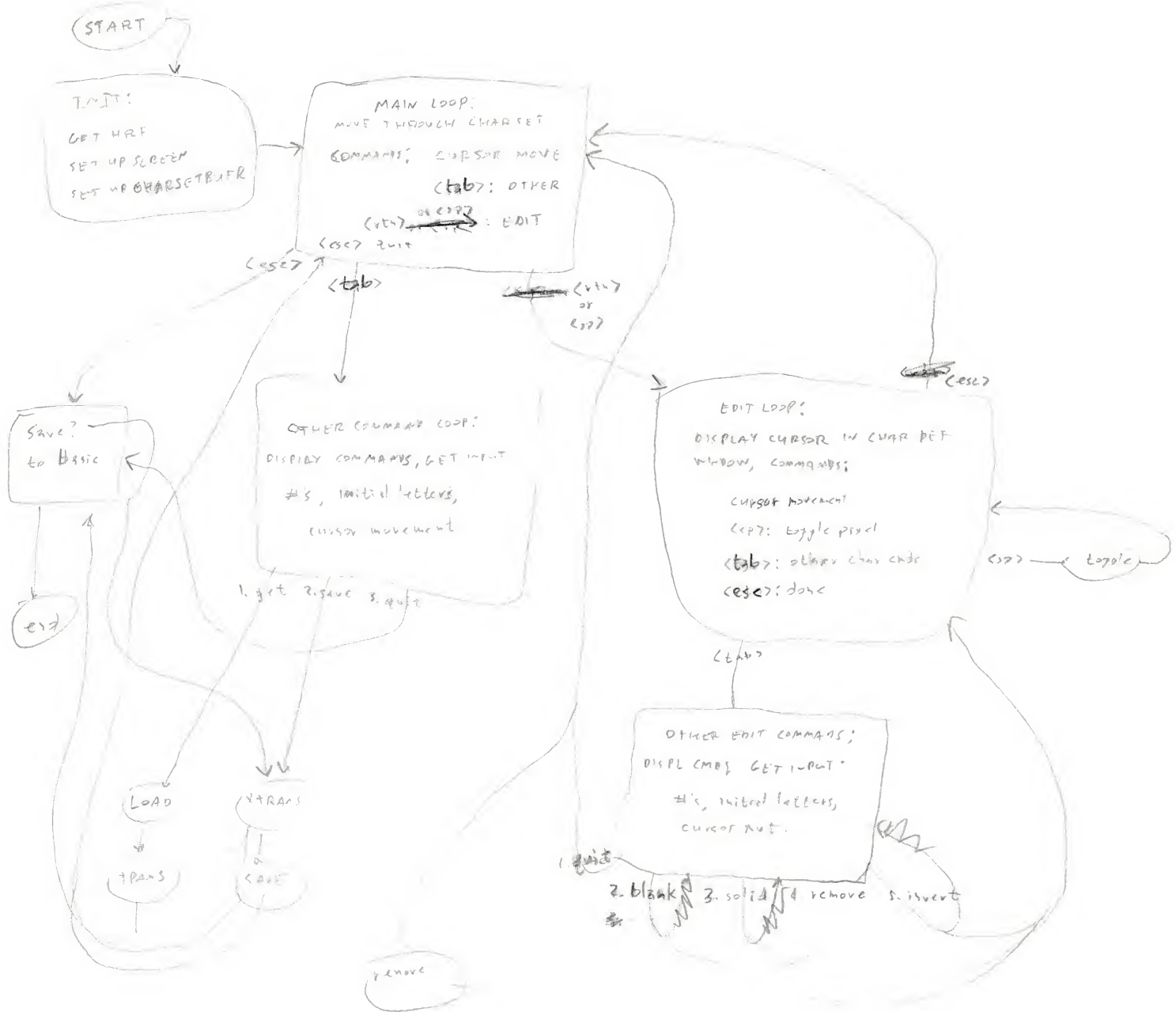


STA PAGE 2

0 ≤ Y ≤ 29, MAIN AND AUX



FEN



9220

Ly. 2/28/20

(amount open 1982)

FED

WORKING  
SET  
BUFR

Hiresp1

GLOBAL + BL

FREE

← FREM EM

H R F

TEXT 02

VECTS.  
GUYEN BF  
STARK  
2 PG

WORKING SET BUFFER:

44 222

\$4.45; ~~\$7.00~~

6 9 0 F 8 i: b b b b b b b b b b b b

40E8: FF FF FF FF FF FF FF FF

4400

4200:

4A\$φ:

52b bytes

BITMAP OF ~~CHAR~~  
DEFINED CHARS

312 bytes  
HEADER

\$100 bytes for  
COMPARISON  
TABLE

(max len)

5803 4-12-65  
CHARACTER

## DEFINITIONS

(max len (256  
chars))

\$4\phi\phi\phi - \\$4\phi 1F:

\$4\phi EE - \\$4\phi FF:

+41¢¢ - \$41FF:

542FF - 549FF:

BITMAP

HEADER

### COMPARISON TABLE

### DEFINITION TABLE

- FOR 1/2

FOR PROG  
WORKING

$$1.023 \times 1000,000 \text{ ct/sec} \times \frac{1 \text{ sec}}{60 \times 60} = 1.023 \times 1000,000 \text{ ct/sec} \times \frac{1}{3600} = 284.166 \text{ ct/sec}$$

$$17,050 = 424 \text{ A}$$

$341 \times 509$

$11 \times 31 \times 5$

$11 \times 31 \times 5 \times 7 \times 2$

~~110 x 155~~

$$152 = 7 \times 8$$

# FED

## Character set

datatype     $\phi 7$     (ignored in 4 $\phi$ /8 $\phi$  cols)     $\leftarrow$  (CHRADDR)

section  
bitmap    E4    7 6 5 4 3 2 1 0  
                 1 1 1 0 0 0 0 0

section    sec 0 1 2 3 4 5 6 7  
base        $\phi\phi$   $\phi\phi$   $\phi\phi$   $\phi\phi$   $\phi\phi$   $\phi\phi$   $\phi\phi$   $\phi\phi$   
offsets    off 2 3 4 5 6 7 8 9

section    sec 0 1 2 3 4 5 6 7  
lengths    FF FF FF FF FF FF FF FF  
(-1)

## Comparison

table

section

4 $\phi$  41 42 43

$\leftarrow$  (CMPADDR)

section 5

section 6

section 7

## definitions

FF FF FF FF FF FF FF FF  
22 22 22 22 22 22 22 22

FED

(DEFADDR) → 00 01 02 03 04 05 06 07  
 ↑  
 CURLIN

(defaddr) → 07  
 76543210  
 E4 (11100100) section bitmap  
 0 1 2 3 4 5 6 7  
 00 00 00 00 00 20 40 60 section base offsets  
 FF FF 1F FF FF 1F 1F 1F section length minus one  
 ,A ,B ,C ,D ,E ,F ,10 ,11

,12  
 ,1

,12

} section 3

(cmpaddr) →

HRF

2171  
578  
714  
1039  
1037  
1040

LOAD FILE \$800 → \$15EE

COPY CODE TO MBSR02 → \$D000 → \$D0FF

20 cols.  
(same versions use AX BS02 also)

16 cols  
is illegal

INIT:

WINDOW SIZE

DISPLAY SW'S

COPY GLOBAL TABLE TO JUST UNDER \$2000

FIX ENV. ~~to setup~~ ~~to setup~~ ~~to setup~~

CLEAR SCREEN

SET ADDRESS OF PTRS (to START PROG AT \$4000)

FOR OUTDEV (SETUP ~~to setup~~ ~~to setup~~ I/O VECTS (PRODOS or DOS 3.3))

(FOR PRINTER:

→ BE30 - BE33

STORE USER VECTORS

→ KOUT2 → CEMIN1

IN

3678

~~to setup~~  
to INSTALL OWN P/O DEV  
SET IFE2 - IFE5 (OUT, IN)  
JSR 31FFA

MM 0000 → \$B000 152E - 1616

AX 0000 → ALTP02 1617 - 1671

LDA \$D100

LDI H0AC

BEQ

EVA IFE33

LDI #FFF

BEQ

JSR HOOK 2F2

~~to setup~~

H00K PROD

JSR H00K 2F2

LDI H00C 03P3

LDI H00C

LDI (GRP00) Y

STA \$36, Y

STI

STI

JMP 33EA

PRO

DOS

→ 36

→ BE30

PTS

PRO

APP

→ 3BE30

PTS

3.3

---

→ \$36

JMP 33EA

H00K 2PG

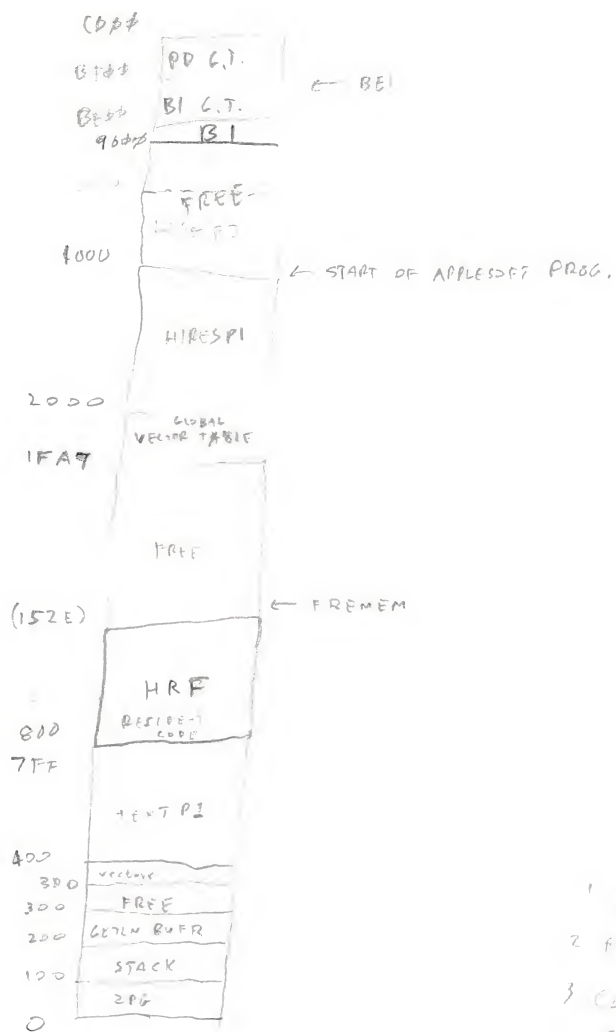
H00K PROD

H00K DTPT

( ~ (P1=76) ) ~ ( ~ (P2=64) )

H R F

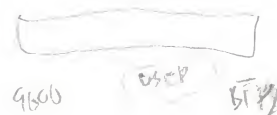
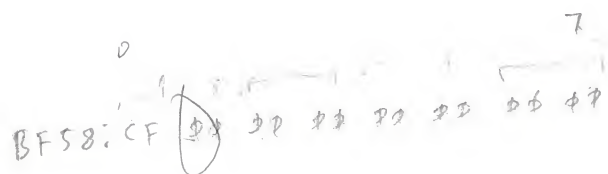
memory map



W BITMAP:

USED PAGES.

0, 1, 4 - 3F, 76 - BF



- ```

1 LINK
2 FILE RESTART
3 CONT
4 PED
5 SYM

```



# GLOBAL VECTOR TABLE

\$1FA9:

10 12 14 16 18 1A 1C 1E 1D 112

FMTADDR 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

FAY. = 8105

IFBD > WHEREMEM  
IFBE

FAY. = 8125

IFBF CHARARG

IFC > FREMEM  
IFC1

IFC2 ID

IFC3 > ESCCURSR  
IFCA

IFCB > ESCCURV  
IFCC

IFCD SSCOSIB

IFCE INVERIO

IFCF UNDERSCR

IFCG PAGE

IFD1 > GTLNCHK  
IFD4

IFD5 > BELVEC  
IFD6

IFD7 > KEYPRVEC  
IFD8

IFD9 > TBLIOVEC  
IFDA

IFDD > DELAYS  
IFDE

IFDF JMP CYCLE

IFE2 JMD SEARCH

IFE5 JMD EXTCELRV

IFE8 JMD DEFADR

IFEB JMD ENVIRON

IFEE JMD BOOTUP

IFF1 JMD BLSTCHR

IFF4 JMD OUTDEV

IFF7 JMD EXTESCE

IFFA JMD EXTESCE

IFFD JMD EXTESCE HMY. = 8184

A1: 7000

A2: 7371

A3: BE DD

A4: 4000

10AC  
152D

L\$ 492

L1170

LDA #0  
STA A2L 3C  
STA A2L 42  
LDA #49  
STA A2L 3E

LDA #7D  
STA A2H 3D  
STA A2H 43  
LDA #44  
STA A2L 3F

SEC  
JMP \$C311

Small

original

bold

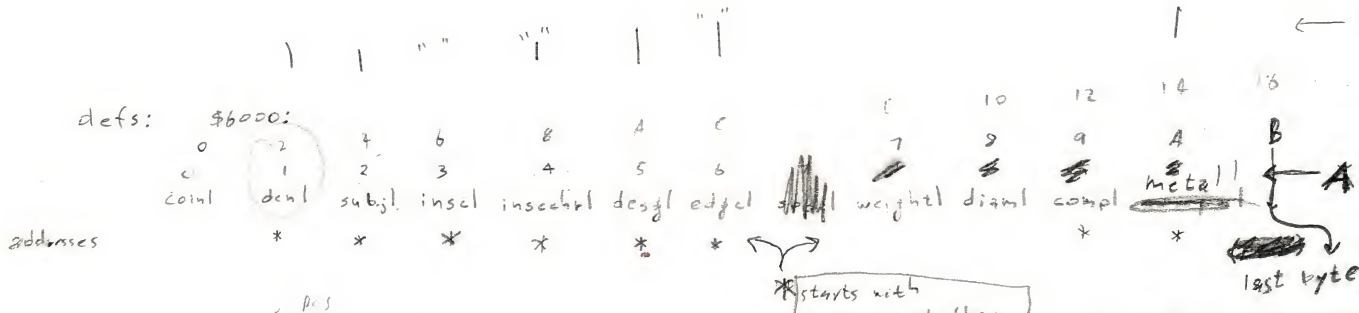
text

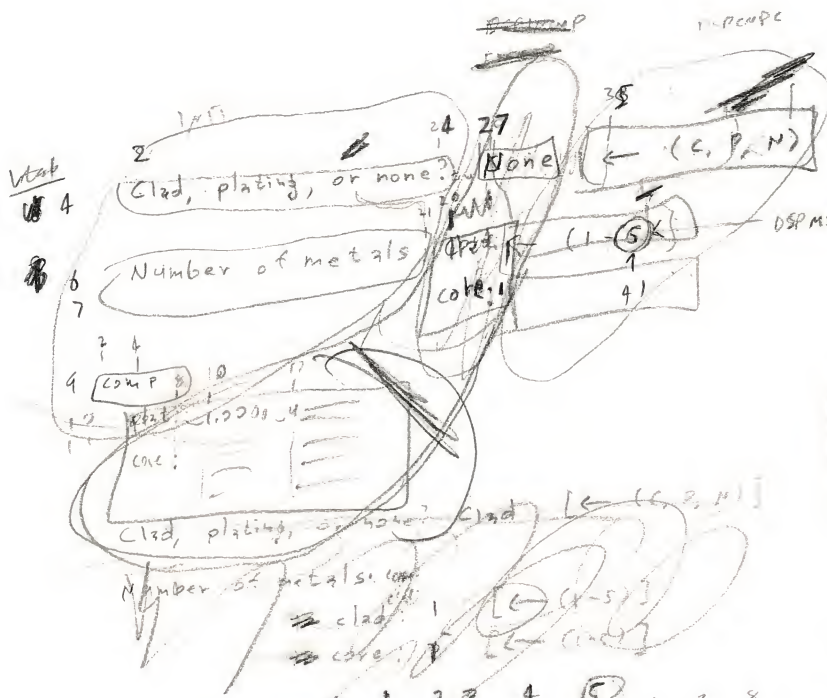
uncial

irish

# Coin definition

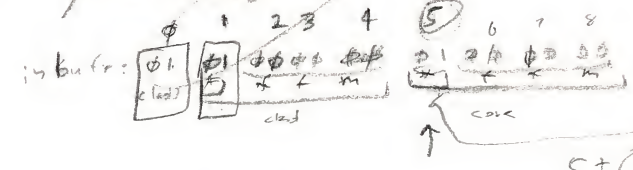
data blocks of normal strings





55  
b - composition:  
1.0000 unknown

DSPMXXEMP  
[2]  
plating  
3  
4  
5  
6  
7  
8  
← # of metals + 2  
compos, + 3:  
→ clad: 1.0000 unknown  
→ core: 1.0000 unknown  
→  
→  
→  
→  
→  
→  
→



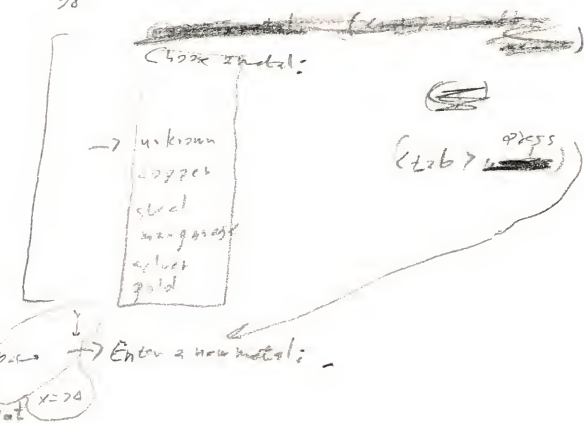
ex: C 13 (core) 0000

inc tab  
if tab = #metals + 10,  
vtab := 10

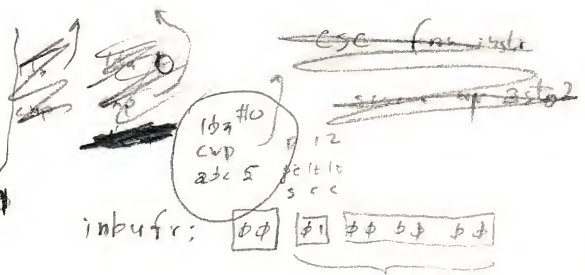
dec tab  
if vtab = 9,  
vtab := #metals + 7

asl  
adc inbuf + 1  
adc #2  
jmp first

692



Enter a new metal;  
input x=70

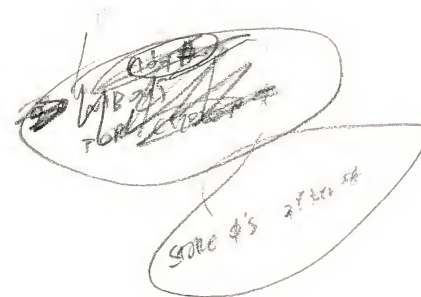


try  
lda inbuf, y  
pla

623  
668  
688

"core:"  
pla  
jst without

1F82  
11100100



# Memory consumption

| field | max string |
|-------|------------|
| 0     |            |
| 1     |            |
| 2     |            |
| 3     |            |
| 4     |            |
| 5     |            |
| 6     |            |
| 7     | \$2FE      |
| 8     | \$1FF      |
| 9     |            |
| A     |            |

appt pps.

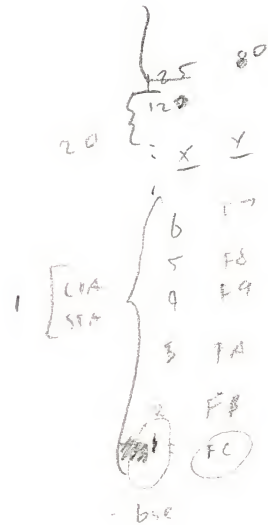
3  
2

esc from input

denom

working for adding

den  
sub  
incs chr ("")  
desph  
edge ("")  
wt  
metal



$$Y = FD - X$$

$$Y = 253 - X$$

965

384  
392

302-312

A9 FF      ldr #1FF  
 A0 00      ldr #0  
 99 00 03      sta \$p3+0, Y  
 C0      iny  
 C0 10      epy #16  
 D0 FD      bne  
 60      rts

00HGT = 8A

OLD TOP  
BIN  
LFT } OK

A 10100101  
10100101  
10100101  
X=8A

(COPY START) → DEY

2466, Y  
6400, Y

2066 DOOU  
2165 DUFF  
2166 b100  
2265 DIFF

2466, Y

~~AD 94 94~~  
~~24~~  
DOADL 1B7  
CD COM  
EF MENU  
G MONITOR  
H1 ZERO MEN  
TL BLOCKAP

C-D  
E-I  
J-L

~~WHD~~  
 $X \leftarrow (WHD BTM - WHD TOP - 1) * 2$   
 $X \leftarrow (X * 2) + X$   
 $A \leftarrow 58A - (WHD BTM - WHD TOP - 1) * 6$   
 $58A - 6(20000 - 1) * 6$

22 23 24 21  
28 27 26 25

21BD  
2165  
2166  
2000

2189  
21A7  
2067  
2060  
HYPER-TEXT - CROSS  
HYPER-INTELL  
L358  
L358  
L3158  
L3159

LHA RAM RAND2 } DK  
LHA RAM RAND1 }  
[ ~~NO SRG HERE!~~ ]  
LHA RAM RAND1  
LHA RAM RAND2  
[ ROM STUFF ]  
6F b265 27

FC76C

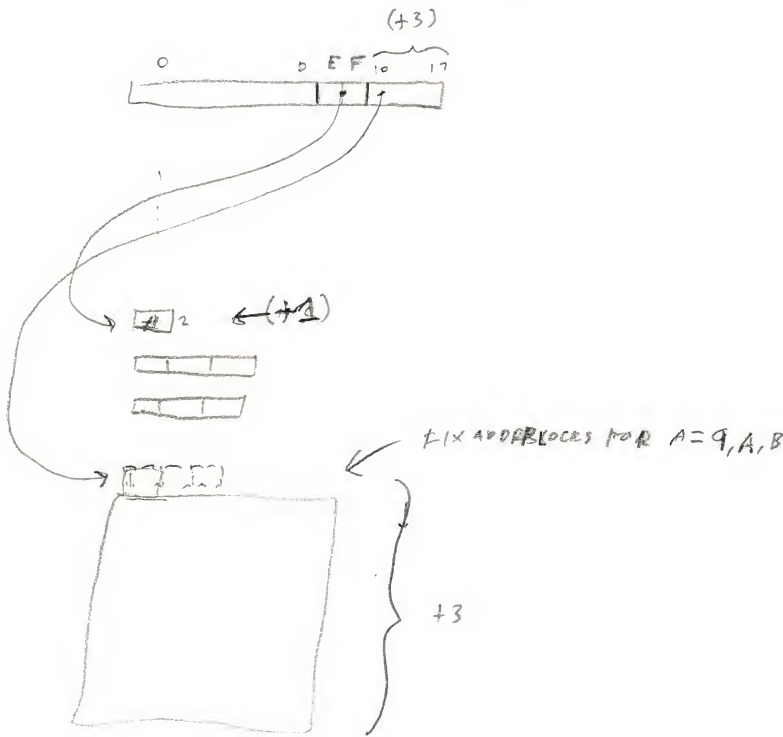
RAM RAND1  
RAM RAND2  
RAM RAND1  
RAM RAND2  
RAM RAND1

ESC from inwt



→ FC: E2 E5  
FD: F1 A0  
FE<sup>tr</sup>: F0 F2

addtent



- ✓ 1. move up block by 3 bytes
- ✓ 2. inc # of bytes
- ✓ 3. get defs + 10, 11 and store new wt
4. ~~inc~~ inc addr in header by 3 (defs + 12 - 10) 12
5. fix addr blocks (+3) (defs + 12 - 10)

inc add  
i: (addi) → start on block (len)  
addr 2 (inc 4)

~~add~~

addtent 2EF4  
invt 2F84  
1: 85b, -2, -3  
E2-E4

Many address  
coming up back in

2FB0 is 47

fix (2)  
[M]

should be: 2  
is actually: 9F + 2

00010111  
00101110  
01011100



ihw

i: b3, y  
o: a30, -2, -3

a30 a302 a303

$y \leq 71$

00 00 00 0

str htab

print mess

h3 #

sta a30

sta a302

sta a303

h3 htab

jir dsp 2p4

jir waitkey

cmp # \$80 ; <return>

beq rts

cmp # \$FF ; <delete>

beg

cmp # "a"

cmp # "q"

plb

cmp # "q" + 1

bge

asl

asl

asl

asl ; forget high nibble

~~...~~

pmc slast  
pmc slast  
pmc slast  
pmc slast

? pf1s1

? invs1

PEO:

flaky  
kversion  
rust  
m1  
b.thap  
machid

add utent

10p:

key  
cert  
paper  
paper  
eternity

562

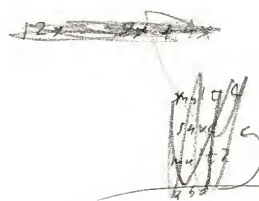
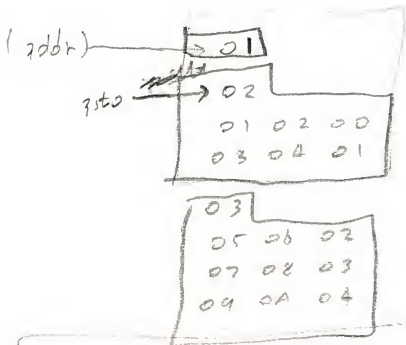
646

size

: a30 a302 a303  
i: ,a 1 bits

mac  
asl  
rol a303  
rol a302  
rol a30  
com

X cur  
X org



in wt

in dm

in comp

in inser

input

00.0000

↑ ↑ ↑

asto -2 -3

input

00.00 mm

↑

asto -2

dsp 2p2

display 2 dgs "." 2 dgs

in asto, asto2 bed, bss, y

as "aa.bbb" where aa = asto, bbb = asto2

skip

y inc, bss pres

lda asto

jsr out

lda asto2

jsr out

rts

jsr nibout

fall through

EE2

VIA8 EB

BASIC 20EF

E2 E3 E4

00 00 00

1

has 457 labels

dsp 2p4

jsr dsp2p2

lda asto3

jsr out

rts

dsp 1p4

lda asto

jsr nibout

lda asto2

jsr out

lda asto3

jsr out

rts

prfld

Q = 9 or 19

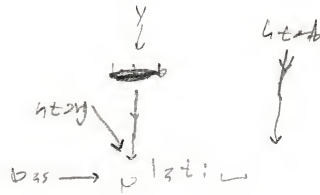
00081001

where  $\bar{x}=0 \leftrightarrow \text{Vertical}$   
 $\bar{x}=1 \leftrightarrow \text{Horizontal}$

as base hnd

ladd: pfq-

X, base Y



asl  
 asl  
 asl  
 pha

; remember (in bit 7) if var H

a = X1001000

lda #9  
 sty hndb  
 sty hndb or f  
 jsr pntastor

ldy #0  
 lda (addr), y

iny  
 cmp #0  
 beq

plz ← ; var H in bit 7  
 jsr pntcmp  
 rts

beq  
 jsr pntcmp  
 asc "plgt: "  
 dcb #

jmp  
 jsr pntcmp  
 asc "clad: "  
 dcb #

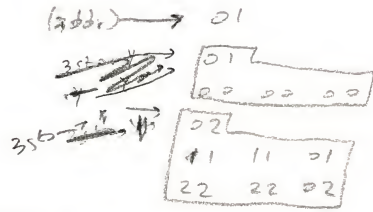
sty hndb  
 cpy #80  
 beq  
 ldy asto  
 plz  
 pla bpl ; (vert)  
 jsr pntcmp

; (horz)

jsr pntcmp  
 sty asto  
 ldy scab  
 cpy #80

beq  
 jsr pntcmp  
 asc "; core: "  
 dcb #

lda #20  
 sta vtab  
 jsr base/c  
 ldy #0  
 ldx #0  
 lda #819  
 jsr pntfld  
 brk



SK2

SK3

SK5

SK6

plz  
 rts

plz  
 rts

# prfunt

i: addr, y, bas, htab  
 n.pres: addr, bas  
 inc: y, htab  
 dest: asto, x

(addr) f1 f2 m  
 asto

htab  
 1.0000 metal 1

ihv  
 lda (addr), y  
 pha  
 dey  
 ora (addr), y  
 bnc  
 lda #"  
 bne  
 lda #"  
 sty asto  
 ldy htab  
 jsr cont  
 sty htab  
 cpy #80  
 beq  
 pla  
 ldy asto  
 jsr cont  
 sty htab  
 cpy #80  
 beq  
 ldy asto  
 lda (addr), y  
 iny  
 iny  
 sty asto  
 ldy htab  
 jsr cont  
 sty htab  
 pla  
 cpy #80  
 beq  
 jsr cont  
 sty htab  
 cpy #80  
 beq  
 lda #80  
 jsr cont  
 sty htab  
 cpy #80  
 beq

## prcomp

i: addr, y, bas, htab, a(x.....)

o: horiz: pres: addr bas vtab  
 inc: y htab  
 vert: pres: addr htab  
 inc: y bas vtab

0 = V  
 1 = H

dest: asto, asto2, htaborg, x

htaborg  
 asto2  
 03  
 01 02 00  
 03 04 01  
 05 06 02  
 0.0102 metal 1  
 0.0304 metal 1  
 0.0506 metal 2  
 0.0102 metal 1  
 0.0304 metal 1  
 0.0506 metal 2

21F6 → htab E5  
 21EF → vtab E0  
 21FC → A  
 21F3 →

after 1st loop y not right

end (y not right)

251

~~251~~

lda htab

sta htaborg

lda (addr).y

sta asto2

iny

pcr

jcr pcrnt

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

;(vert):

~~lda htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

jcr pcrnt

lda htaborg

sta htab

lda utab

jcr bascale

(horiz) jcr pcrnt

sta asto

lda asto2

cmp #1

bca

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

~~sta htab~~

dec asto2

bne pcrnt

lda #5A#

jcr cont

sta htab

cpy #80

bca

lda asto

~~sta htab~~

~~sta htab~~

~~sta htab~~

bcs

C=X  
1/2

dec asto2

bne pcrnt

iny

iny

iny

iny

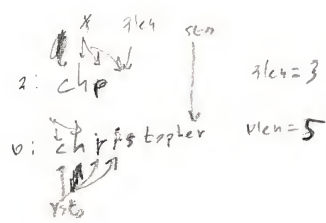
iny



~~in~~

$x, b, aLen, bLen$

$bLen \geq aLen$



Equal: 1

$a \leq 200$

$b \leq 300$

$aLen \leq FE$

$bLen \leq FF$

Equal: FD

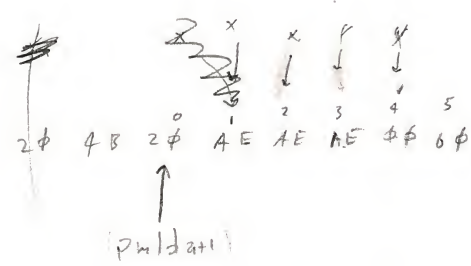
\$4000

$a = 9$  or  $17$

$000 \times 1001$

$x \ 1001000$

when  $x = 0 \rightarrow$  Vertical  
 $x = 1 \rightarrow$  Horizontal



$x \ 1001000$

(addr)  $\rightarrow 01$   
 $\rightarrow 0.2$   
 $\rightarrow 11 \ 22 \ 33$   
 $\rightarrow 44 \ 55 \ 66$   
 $\rightarrow 02$   
 $\rightarrow 10 \ 20 \ 30$   
 $\rightarrow 40 \ 50 \ 60$

AE

$A = \emptyset$



2447

forward

SDSD

print of below line

when anything before

forward:

screen up

+ including  
 $0.0000x$

runs off

address test

i: 7  
in b, fr

2C9A

~~ADDSPEC~~

ATTN: PERKINS  
26 FEDERAL PLAZA  
NY NY 10278

bufr: C H R  
C3 C8 D2 44

~~1st 1st 1st~~  
~~1st 1st 1st~~

6083.60FF

fn bld d d  
back

713A  
608C

change: ||

don't inc  
blocks: 1=7,8  
(1=F, 11, 11)

x=12X

ex: 68 68 AA 34 64  
bmt: 68 AA 68 18 64

3.2.2  
scheduling

last  
↓  
61F8: F2 44 A5

1265, 1267

1296, 1303

architect  
start up cycles

sta page2  
ldy #37  
[ldy sca pairs]  
dec  
bne  
lda page2?  
bpl cc  
sta page1  
bmi cs  
rts

4  
2  
X  
2  
2, 3, 2  
4  
3, 2  
4  
3

4 4  
2 2  
2 2  
3 3  
1 1  
2 4  
2 4  
4 3  
3 3  
main loop  
2 2  
3 3  
2 2  
3 3  
24  
HLS 28

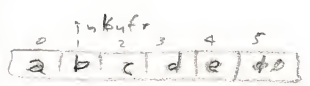
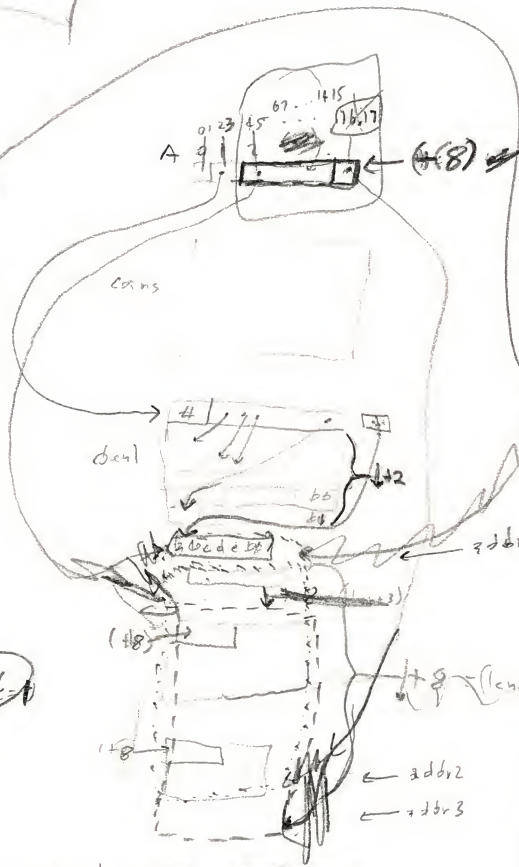
~~xxxx~~  
~~xxxx~~  
low x  
x only

~~xxxx~~  
2563 = constant  
~~xxxx~~  
~~xxxx~~  
xxxx



add inputs to database  
i:  $2(1-A)$   
inputs  
o:  
dest:

addr3 ← len+3

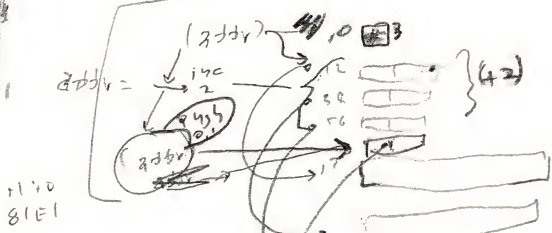


A hand-drawn diagram consisting of a vertical line. On the left side of the line, there is a circle containing the number '2', with an arrow pointing to the left. Above this circle, there is a horizontal line with a vertical tick mark in the center. Below the circle, there is a horizontal line with a vertical tick mark in the center. At the bottom of the vertical line, there is a small 'v' shape. To the left of the bottom of the vertical line, there is a small '4.3)'.

$(y-1) + 2$  1. MOVE ~~\_\_\_\_\_~~  
 $(A(2+1)) \cdot (A(B))$   
 $2 \times 2$   
 $\rightarrow (A(2B)) + 3 + 1 \text{ en } \checkmark$

251  
tay  
lda defs, y  
sta addr  
ldy  
lda def, y  
sta addr + 1  
ldy # 0  
lda (addr), y  
~~\_\_\_\_\_~~  
~~\_\_\_\_\_~~  
sta ~~\_\_\_\_\_~~ asl ~~\_\_\_\_\_~~ rsl  
~~\_\_\_\_\_~~  
lda addr + 1  
adc # 0  
sta addr + 1  
~~\_\_\_\_\_~~  
lda addr  
qld rsl  
sta addr  
~~\_\_\_\_\_~~  
lda addr + 1  
adc # 0  
sta addr + 1  
~~\_\_\_\_\_~~

831  
 set 75to  
 pto  
 clc  
 loc addr  
 alc 25to  
 sta addr  
 loc 33dri  
 alc 45  
 pto  
 ads #0  
 clc 33dri

$$\begin{array}{r} +1 \quad +0 \\ 2dfy = 7FF F : F8 \\ 8 \quad F1 \\ c = \cancel{F} 1 \end{array} \quad \begin{array}{r} 2 = \cancel{F8} \cancel{F8} \\ 2560 = E2 \\ \hline E1 \end{array}$$


more (savings, etc.)

```

101 // 0
102 addr
103 str(addr), y
104 my
105 102 addr + 1
106 str(addr), y

```

prevent, moved

1d7 addr  
 adc #0 f 2d7 1  
 sta addr  
 1d7 addr+1  
 adc #0 : clear C  
 sta addr+1  
 jmp  
 1d7 (addr), y  
 adc #2  
 sta (addr), y  
 iny  
 1d7 (addr), y  
 adc #0 : clear C  
 sta (addr), y  
 1d7 addr  
 adc #2  
 sta addr  
 1d7 addr+1 : clear C  
 adc #0  
 sta addr+1  
 dec bnc  
 1d7 addr

## update

1. display current definitions [~~at bottom~~, <sup>obj</sup> ~~subject~~]
2. ~~press~~ use ~~arrow~~s to move thru defs.;    ↑: \$88 or \$8B    ↓: \$8A or \$95  
      <return> to edit a def.;                \$8D  
      \* A \* to add a def.;                \$C1  
      <esc> to quit.                        \$9B

<return>:  
edit a def.

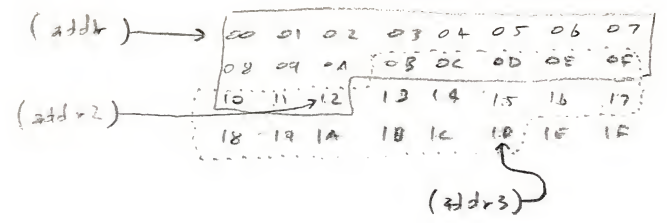
<arrow>s move through fields  
<return> to work on a field  
~~At add a new one to the current field~~  
~~to make~~

<return>:  
work on a field

<arrow>s to move thru current choices  
A to add a new choice

~~addr - addr2~~  
 move back up \$B bytes

\$264D  
 move up B



```

clc
lda addr2
adc #B
sta addr3
lda addr2+1
adc #0
sta addr3+1
  
```

$addr3 := addr2 + B$

addr: ~~01~~ 04  
 addr2: ~~02~~ 04 0F 04  
 addr3: 01 20 ~~FE 1F~~

\$26AD  
 move up

~~addr - addr2 → addr3 -~~

```

ldy #0
lda (addr2), y
sta (addr3), y
ldx addr2
cpx addr
bcc
flex
cpx
  
```

```

ldx addr2
bne
dec addr2+1
cpx addr
bcc
  
```

0402 → 2001 ✓  
 0401 → 2000 ✓  
 0400 → 1FFF ✓  
 2601 → 1FFF ✓

```

ldx addr2
bne
dec addr2
ldx addr3
bne
dec addr3+1
dec addr3
bne (bra)
  
```

addr \$F2-F3  
 addr2 \$F4-F5  
 addr3 \$FC-FD

r: FE 7F  
 r2: FE 7F  
 r3: 00 80

A = 14  
 X = FF

7FFF → 8001 ✓  
 7FFF → 8000  
 7FFD → 7FFF  
 7FFC → 7FFE

|  | → 3B  | 55    | 88    | 20    | 40    | 71    | 7<br>Lt | 8<br>diam | 9<br>comp | A<br>comp | B<br>last |       |
|--|-------|-------|-------|-------|-------|-------|---------|-----------|-----------|-----------|-----------|-------|
|  | 18 60 | 30 60 | 4A 60 | 7D 60 | 85 60 | 42 61 | 66 61   | 97 61     | 9E 61     | A3 61     | C3 61     | F9 61 |

6018: 02 00

601A: 0 0 0 0 1 1 0 0 0 0 1

6025: 1 2 2 1 3 3 1 0 1 1 2

6030: 2 35 60 3A 60

6035: E3 E5 EE F4 φφ

603A: F6 ... E5 φφ

604A: 4 53 60 5B 60 68 60 72 60

hundreds: F8, F9

columns: FA, FB

coin: F6, F7

add a coin:

is coin

~~270F:AD~~

addcoin \$26F8

```
lda defs+32
sta addr
lda defs+33
sta addr+1
lda defs+34
sta addr2
lda defs+35
sta addr2+1
jsr moveupB
```

move rest of data up

270F:AD

```
ldy #5A
lda (coin), y
sta (addr), y
dex
bpl
```

init new coin  
to current coin

2718:A5

```
lda addr
sta coin
lda addr+1
sta coin+1
```

addresses:  
current coin =  
new coin

```
lda defs
sta addr
lda defs+1
sta addr+1
```

inc  
number  
of  
coins

```
ldy #0
lda (addr), y
adc #1
sta (addr), y
```

273A:A2

inc addr

add \$B to addr

i: addr: address of address

o: address inc by \$B

dest: y, z

```
clc
ldy #0
lda (addr), y
adc #B
sta (addr), y
```

```
ihy
lda (addr), y
adc #0
sta (addr), y
rts
```

```
lda $FFFF, x
sta $FFFF, x
inc x
```

routines used by addcoin

inc addrB i: addr add \$B to address

dest

y, z

moveupB i: addr, addr2 move up block \$B bytes

x, y, z, addr2, addr3

inc allB i: addr inc all addrs in address block  
(often bypassed to by addr)

ldx #58B

```
lda defs, x
sta addr+1
dex
lda defs, x
sta addr
dex
jsr incaddrB
lda #1
bne
```

fix  
initial  
addr  
block (+B)

2753:A2

276C:

addrsti: 010203040506070A

A in defs of data blocks  
with addrs to  
fix (+B)

```
ldx #0
lda addrsti, x
asl
tay
lda defs, y
sta addr
iny
lda defs, y
sta addr+1
ctx rto
jsr incallB
ldx rsto
dex
bpl
```

rts ← 276B



incabB

i: addr

ldy #0

lda (addr), y ; # of addresses

asl

sty asto

inc asto

; first offset  
; first ~~addr~~ let used

ldy #1

~~ldy #1~~

clc

lda (addr), y

adc #B

sta (addr), y

iny

lda (addr), y

adc #B

sta (addr), y

iny

cpy asto

bne

rts

(addr) → 05, 0

, 1, 2

00 01

, 3, 4

F.F 02

, 5, 6

80 38

, 7, 8

61 52

, 9, 10

F.F 00

, 11, 12

XX

low value

(start)

0B 01

0A 03

8B 38

6C 52

09 01

incaddrB: \$26D4

i: F2-F3

(addr) → 18 60

incabB: \$26E5

i: F2-F3

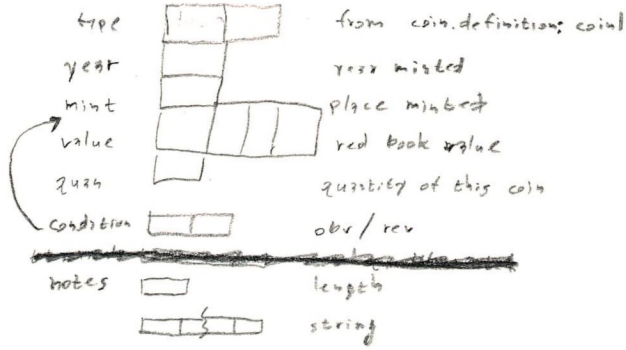
(addr) →

~~incaddrB~~

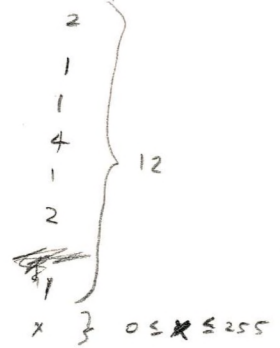
# COIN

coin data

each coin:



# notes





print comp list

i: lab, bas, addr, y,  
z (# words)

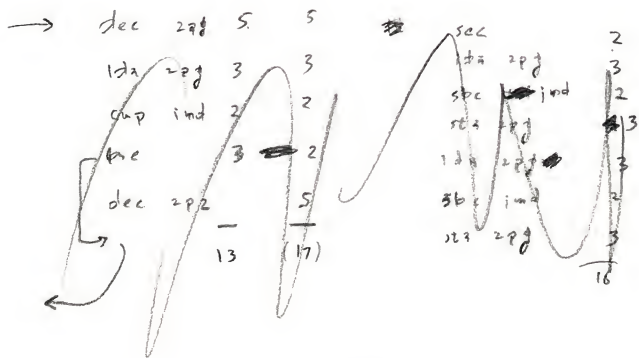
2305:

Print Metal

i: 2 (metal#), 45, 2503 (416)

242 F:

lay  
lda HHA  
jrr petaddr  
ldy acts3  
sr prstr  
rts



|      |         |    |
|------|---------|----|
| idx  | coinson | 3x |
| dex  |         | 2x |
| cp x | Niff    | 2x |
| bt x | coinson | 3  |
| bnc  |         | 3  |
| dec  | coinson |    |

21E4

addr: F2.F3

1930-1931 20.75

hypothesis:

02 40

6717544.

2.6. Fi

98

03

2

назад

4

Can't see      Can't see

01 46 ... 20 00

---

11.

ok

$$1/2 \quad \text{anisotropic}$$
$$\cos p \quad \cos q \sin p + 1$$

bcc ~~refined~~

one ok next  
1/23 ~~quarter~~

( July 24-25 )

\_\_\_\_\_ eye

10 25 ~~42~~ ~~100~~ —

ex46:

→ jmp loop

$$u_{\ell}^k| :$$

inc kingdom

bne

inc 2045461004

ndc # 3B

COIN

files

|                 |                |                                                |                             |
|-----------------|----------------|------------------------------------------------|-----------------------------|
| prodes          | SYS            | Prodes system                                  |                             |
| coin.system     | SYS            | main rxu/vcl/view system ( $\Delta$ coin.data) | A\$2000                     |
| coin.update     | <del>SYS</del> | update utility ( $\Delta$ coin.defn)           | A\$2000                     |
| coin.data       | BLN            | list of coins                                  | A\$6000 (after coin.system) |
| coin.definition | BLN            | coin/year/... data                             | A\$8000                     |



# display a coin definition

1 0.175

cent 2

4 Abraham Lincoln

7 "IN GOD WE TRUST LIBERTY (date) (mm)"

8 by Victor D. Brenner

10 edge: "E Pluribus Unum"

13 wheat stalks

15 "E. PLURIBUS UNUM ONE CENT UNITED STATES OF AMERICA"

16 by Victor D. Brenner

weight 3.11 g

diameter 19 mm

composition ~~brass~~ ~~brass~~ ~~brass~~

.9500 copper, .0500 tin and zinc

20  
21  
22  
23

addr → 0

2  
00 75 00

01 25 00

1  
00 00 00

350 = 00 (metal)

13  
↓  
h2b  
↓  
1.0000  
↓  
copper

13  
↓  
h2b  
↓  
1.0000  
↓  
copper

inc 5  
152 (355) 5  
sta (355) 6  
16

inc 5  
inc 5  
sta 5  
4  
20

addr → 0 350

2  
00 00 00  
01 34 56  
3502

1 → 01

mul2  
sta 5  
inc mul2+1  
inc mul2+1  
inc mul2+1  
inc mul2+1  
inc mul2+1

2  
0  
34  
56

2

350 = 01 (metal)

350 = 01  
1st metal

1  
1

1.0000  
↓  
copper

upmove  
have look up 38 bytes

start  
5  
3502  
5

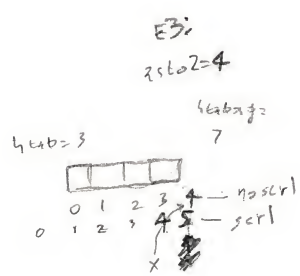


instr  
 vtab ← (scrn pos) ≤ 22  
 X (last col to display) ≤ 78  
 as: htab = string (ASCII \$A0-\$FF)  
 (term - / 00; no \$80)

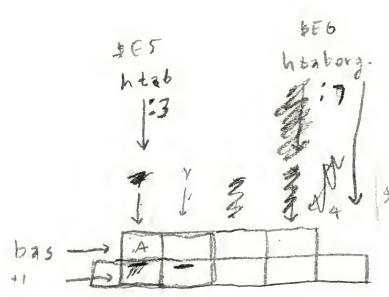


key send functions

- ASCII \$A0-\$FF (00) ~ enter character
- \$88 (left-arrow) X SEQU
- \$95 (right-arrow) back-nuclear
- \$FF (delete) back-clear
- \$9B (esc) quit - input empty
- \$8D (return) done
- ~~\$80 (tab) insert~~



B L T  
 0 2 00 14  
 0000 0000 0000  
 1 3 2 8



1-72  
 if X (7-31) then scroll it

scrinlt  
 xwrite  
 i: htab, htaborg  
 as: pres: htab, bas, X, Z

input: 0, 1, 2, 3, 4, 5, 6  
 as: (len)  
 \$E2

1336-1407

if Y is even  
 in Aux X ← M  
 if Y is odd  
 in Main M ← X  
 if Y is even  
 in Aux M → X  
 if Y is odd  
 in Main X → M

tya  
 pha  
 tsv  
 tsv  
 bcc  
 sta page2  
 iny  
 tdo (big), Y  
 sta page1  
 dey  
 sta (bas), Y  
 pla  
 tsv  
 rts





2528: prfld

```
ldz #18
sta vtab
jsr bascalc
ldy #4A
lda (con), y
tax
lda #9
ldy #13
jsr prfld
```

print individual field

1: bas, y (vtab, htab)

a: (con)

- 0 den
- 1 sub
- 2 insc
- 3 inschr
- 4 desg
- 5 edge
- 6 weight 17% #
- 7 \*
- 8 diameter 18% mm
- 9 comp 19 cone ←
- A metal

X: which one in list ← don't check for validity

first: addr, ~~h~~ htab

Y = htab for next print

dst

1, 2, 4, 5, A: addr, htab := 7  
 3 addr, htab  
 6 addr, htab  
 7 addr, htab, addr2  
 17  
 18  
 19 → addr, ~~addr~~, htab  
 block, addr, 2, 3  
 addr2

for a=1, 2, 5, 6, A

lbr htab

```
asl
tay
clc
lda defs, y
adc #1
sta addr
iny
lda defs, y
adc #0
sta addr+1
lxa
asl
tay
lda (addr), y
tax
iny
lda (addr), y
sta addr+1
stx addr
```

453, 471

EXAMP: lbr #0  
 lbr #0, y  
 tax

which one

580 - 443

ldz #2

sta vtab

jsr bascalc

ldz #1

ldy #5

jsr prfld

bas

denomination

hbr

ldy #0

ldy (con), y

tax

255  
 1  
 322

465

423 - 450

prfld

546

548

dst

ldz #3A2

jsr cout

lbr htab

jsr prfld

ldy htab

cpy

jsr prfld

bcc

ldy #76

jsr prfld

tax

ldy #77

ldz #3A2

jsr cout

rtz

ldy htab  
 jsr prfld

addr  
 y  
 already  
 it: addr, bas, y

prfld  
 #2, X  
 (ac prfld)

A.D

CALC  
STGPH  
AT 8:00

prfld

print composition (z=9)

i: a = 9  
 X = which one X  
 bas = base addr for print  
 Y = htab for print X (htab)

getaddr(addr) → 1  
 2  
 25 00 00  
 75 00 01  
 3  
 10 00 02  
 30 00 03  
 60 00 04

bas → ctad: x  
 X. 2500 copper  
 X. 7500 nickel  
 X. 1000 steel  
 X. 3000 zinc  
 X. 6000 silver

(z=17)

i: a = 19  
 X = which one X  
 bas = base  
 Y = htab

getaddr(addr) → 1  
 2  
 25 00 00  
 75 00 01  
 3  
 10 00 02  
 30 00 03  
 60 00 04

bas → ctad: x. 2500 copper

print weight (z=7) 22 block

i: a = 7  
 X = which one  
 bas = base  
 Y = htab

(defstE) → 04  
 0: 1 00 00 00  
 1: 4 11 11 11  
 2: 7 22 22 22  
 3: A 33 33 33  
 0 →

1 x a  
 5 + 4 150  
 25 1  
 25 1  
 25 1

X = 80  
 z = 0  
 3500 = 80

X\*3  
 +  
 defstE  
 +  
 1

276 - 310  
 294 - 314

OFF1  
 11-1431  
 G.S.S  
 T.CSP  
 T.CSL  
 C.S.F  
 T.CUL  
 C.D.S

✓ T. PRE EQU  
 ✓ T. PRE START  
 ✓ T. PRE DATA  
 PDOMAIN.S

PREMINS

PREREL.S

2000 - 3FFF  
 2100 - 3FFF  
 4000 - BEFF

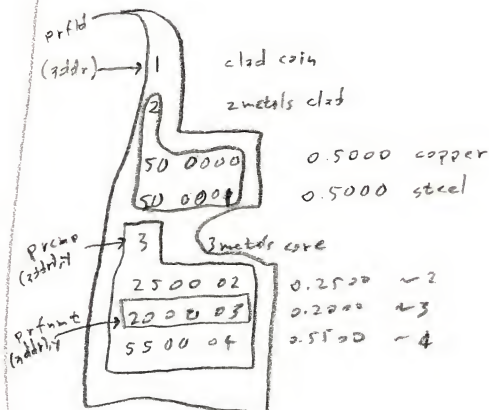
REL  
 MAIN  
 EQU  
 START  
 DATA

PREMINS A050 - AD67

2000 cu start

2050 start to stop PROPS stuff

### composition list



0 1 2 3 4 5 6 7 8 9 A

ecvt: ~~1~~ 2 5 6 7 A D E F 11 14 11

echt: ~~1~~ 0 8 8 8 8 8 8 7 9 17

ecchr: 0 0 0 0 0 0 0 80 80 80

PL M1

prfld

i: a, x  
bas, y

→ 1 class

2

50 00 00

50 00 00

3

25 00 02

20 00 03

55 00 04

prfhnt print fineness and metal

i: (addr), y → ff ff mm

dest: ~~2500~~, x

bas, htab

o: pres: addr, bas inc: y, htab

prcmp

print one list

i: (addr), y → # metals

bas, htab

z: 80 = horiz.

00 = vert.

(vert. htab)

o: pres: addr, ~~bas(horiz), htab(horiz)~~ (horiz: ~~bas~~, bas, htab)

inc: y, htab, ~~bas(vert), htab(vert)~~ (vert: ~~bas~~, bas, htab)

(horiz: htab)

dest: 2500, 2500

htaborp, x

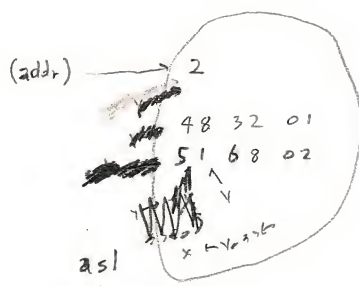




pr cmp

63  
core: 1.0000...  
3-4

2-84



asl  
php  
lda htab  
sta htab  
lda (addr), x

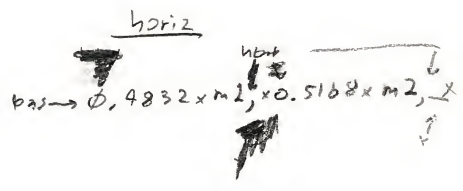
; # of metals (20)

~~beq~~

2

iny  
sta asto2

; # of metals (>0)



dst: asto2  
hstab  
asto  
x

C=1,0(h,u)  
~~~~~

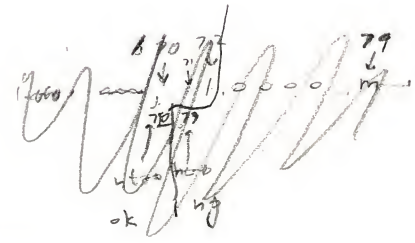
jsr prfmt  
sty asto  
plp  
php  
bcs 3 ; br if horiz  
lda hstaborg  
sta htab  
inc vtab  
jsr bascale  
bcc (bra)  
lda asto2  
cmp #1  
bca  
lda htab  
lda #", "  
jsr cout  
lda #3AF  
jsr cout  
sty htab  
lda asto  
dec asto2  
bne  
pla  
rts

cpy #80  
beq

lda #12  
sta vtab  
jsr bascale  
lda #20  
sta htab  
lda #3AF  
sta addr  
lda #61  
sta addr+1  
lda #0  
lda #0  
jsr prcmp2 269E  
rts

vtab: E0: 12  
(h)bas: F0: 12h  
htab: E5: 20  
baddr: F2: AF 61

61F3, 61DF



69 70 74 75 76 79  
1.0000 m...  
ok hg

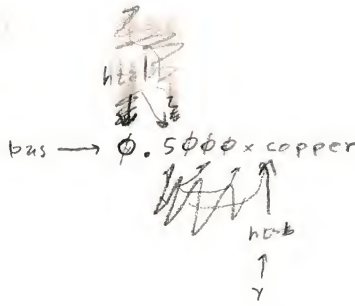
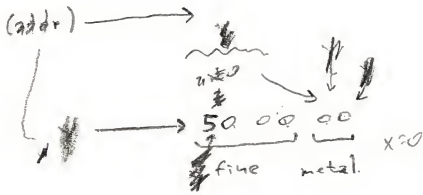
check:

of htab  
bca  
bca  
cpy #70  
bcc  
jsr prmc  
sta htab  
lda asto  
pla  
bne

cpy #78  
bcc  
lda #77

prfmt

~~prfmt~~



addr = F2.F3

bas =

htab =

iny  
lda (addr),y  
pha  
ora (addr),y

bne  
lda # "1"

bne  
lda # "0"

sty asto  
ldy htab  
jsr cout  
lda # ", "  
jsr cout  
sty htab

~~lda asto~~  
~~lda (addr),y~~

ldy asto  
lda (addr),y

iny  
iny  
sty asto

ldy htab  
jsr cout

pla  
jsr cout

lda # \$AF  
jsr cout

sty htab  
ldy asto

lda (addr),y  
tax

lda # \$A  
ldy htab

jsr prfld  
~~lda asto~~  
ldy asto  
iny  
rts

ldy asto, x, y

6'AF: 02

6'BD: 01

6'BL: 00 00 03

6'B4: 01

6'B5: 00 00 02

(uap) ~~E5~~ = 20

(asto) ~~E2~~ = 2

F2: AF 61

Y = 2

21E3: lda #12

sta vtab

jsr pascalc

lda # \$AF

sta addr

lda # \$61

sta addr+1

lda #20

sta htab

ldy #2

jsr prfmt

rts

i:  
4572

lda addr+1  
pha  
lda addr  
sha

pla sta addr  
pla  
sta addr+1

\$6544 ↓

list of addrs of:

|        |      |    |
|--------|------|----|
| coin:  | #:   | 0  |
| den:   |      | 2  |
| inset  |      | 4  |
| inset  |      | 8  |
| desig  |      | 10 |
| weight | spec | 12 |
| diam   |      | 14 |
| comp   |      | 16 |
| comp   |      | 18 |

coin:

# coin types: 2 bytes

02 00

list of coin types;

001A: 00 00 00 00 01 01 01 00 00

each coin type is 4 bytes:

0 000 00 00 01 01 02 00 01

den, ~~side~~ side,

den, ~~edge~~ edge, spec

0h 05 0i 0d rs rir rd ed sp

obv:

# obs: 1 byte

01

list of obs;

each obs is 3 bytes:

00 00 00

sub, inset, desig

rev:

# revs: 1 byte

02

list of revs:

each rev is 3 bytes:

00 00 00

sub, inset, desig

02 00 00

edge:

# addrs: 1 byte

03

list of addrs of edges in order up from 0

edge0 edge1 edge2

list of edges;

ca: string #

0: [plain] #

1: [creeded] #

2: two HUNDRED FOR A DOLLAR #

spec:

# specs: 1 byte

02

list of specs;

each spec 3 bytes:

00 00 00

wt, dia, comp

00 00 01

den:

# den: 1 byte

02

list of addrs;

den0 den1

list of den's;

0: balanced

ca: string #

1: cent #



inse1:

# inse's: 1 byte

6440: 02

list of addrs of inse's in order up from 0

inse0 inse1

list of inse's.

0 1 2 3 4 5 6

ea: lea; insestring

6445: 0: d3

03 A0 00

6449: 1: 06

04 AA 03 04 A0 00

insechr:

# chars: 1 byte

6450:

06 0, 1, 2, 3, 4, 5

list of addrs of chars in order up from 0

6450:

0: ~~00 00 00 00 00 00~~

(3 dynamics)

list of chars;

1:

~~00 00 00 00 00 00~~

ea: string pp

2:

~~00 00 00 00 00 00~~

3:

~~00 00 00 00 00 00~~

4:

~~00 00 00 00 00 00~~

5:

~~00 00 00 00 00 00~~

desgl:

# dgl's: 1 byte

02

list of addrs of dgl's in order up from 0

desgl0 desgl1

list of dgl's;

ea: string pp

0: Adam Eckfeldt 00

1: Robert Scot 00

weigh1:

# weights: 2 byte

02

list of weights;

ea: 3 bytes (aibcc) in BCD:

60B8:

addr → 22 00 00

(22.0000 g)

22.0000 g

21 50 00

(21.5000 g)

diam1:

# diam's: 1 byte

01

list of diam's;

ea: 2 bytes (aabb) in BCD:

addr → 11 45

(11.45 mm)

22.00 mm

subj:

22-141 50 SHEETS  
22-142 100 SHEETS  
22-144 200 SHEETS







Obverses

?

subject:

|            | <u>position</u> | <u>facing</u> |
|------------|-----------------|---------------|
| Liberty →  | head →          | left          |
| Indian     | seated          | right         |
| Lincoln    | standing        |               |
| Jefferson  | walking         |               |
| Roosevelt  |                 |               |
| Washington |                 |               |
| Franklin   |                 |               |
| Kennedy    |                 |               |
| Eisenhower |                 |               |
| Anthony    |                 |               |
| eagle      |                 |               |
| shield     |                 |               |
| star       |                 |               |

designer:

list of designers

# coin. definition

coin:  
 detail  
 obverse  
 reverse  
 edge  
 specifications

obverse:  
 subject  
 inscription  
 designer

reverse:  
 subject  
 inscription  
 designer

specifications:  
 weight  
 composition  
 diameter

edge:  
 { plain  
 reeded  
 lettered  
 otherwise }  
 → string  
 → string

subject:

designer:

inscription:  
 (2 types of ins)  
 string w/ ins.chars, date + mm will be added  
 during use (except poss "P")

weight:  
 3 bytes BCD:  
 xx.yy zz g

diameter:  
 2 bytes BCD:  
 xx.yy mm

ins.chars:  
 3 LIBERTY  
 7 UNITED STATES OF AMERICA  
 5 IN GOD WE TRUST  
 6 E PLURIBUS UNUM  
 4 \* + + + + \* (6 stats)  
 5 IN GOD WE TRVST  
 0 IN GOD WE TRVST.  
 ...

composition:  
~~finchrs~~ - metal





21, 22 Lhl capped classic (2)

27, 28 L seated (2)

39, 40 Indian h (2)

|               |                           |                        |
|---------------|---------------------------|------------------------|
| 0             | Lhl cap & pole            | Adam Eckfeldt?         |
| 1             | Lhr cap & pole            | Robert Scot            |
| 2             | Lhr cap & pole            | John Smith Gardner     |
| 3             | Lhr draped bust, stars    | Scot                   |
| 4             | Lhl classic               | John Reich             |
| 5             | Lhl (braids) coronet      | Christial Gobrecht     |
| 6             | Lhr flowing hair          | Henry Voigt            |
| 7             | Lhr flowing hair          | Eckfeldt               |
| 8             | Lhr cap & pole            | Joseph Wright          |
| 9             | Lhl coronet               | Scot                   |
| <del>10</del> | <del>Lhl coronet</del>    | <del>Scot</del>        |
| 11            | eagle flying              | James B. Longacre      |
| 12            | Indian h                  | Longacre               |
| 13            | Lincoln h                 | Victor D. Brenner      |
| 14            | shield                    | Longacre               |
| 15            | star                      | Longacre               |
| 16            | Lhl coronet               | Longacre               |
| 17            | Lhl coronet               | Charles E. Barber      |
| 18            | Indian h                  | James Earle Fraser     |
| 19            | Jefferson h               | Felix Schlag           |
| 20            | Lhr flowing hair          | Scot                   |
| 21, 22        | Lhl capped classic (2)    | Wm. Kneass             |
| 23            | L seated                  | Gobrecht               |
| 24            | Lhr wreath                | Barber, C.             |
| 25            | Lhl winged                | Adolph A. Weinman      |
| 26            | Roosevelt h               | John R. Sinnock        |
| 27, 28        | L seated (2)              | Wm. Barber             |
| 29            | L standing                | Herman A. MacNeil      |
| 30            | Washington h              | John Flannagan         |
| 31            | Lhl classic               | Gobrecht               |
| 32            | L walking                 | Weinman                |
| 33            | Franklin h                | Sinnock                |
| 34            | Kennedy h                 | Gilroy Roberts         |
| 35            | Lhl coronet, capped       | George T. Morgan       |
| 36            | Lhl crown (peace)         | Anthony DeFrancisci    |
| 37            | Eisenhower h              | Frank Gasparro         |
| 38            | Susan B. Anthony h        | Frank Gasparro         |
| 39, 40        | Indian h (2)              | Longacre               |
| 41            | Lhr capped                | Scot                   |
| 42            | Lhl classic, capped       | Reich                  |
| 43            | Indian h                  | Bela Lyon Pratt        |
| <del>44</del> | <del>Lhl curly hair</del> | <del>Barber, C.</del>  |
| <del>45</del> | <del>Lhl curly hair</del> | <del>Barber, C.</del>  |
| 46            | Indian h                  | Augustus Saint-Gaudens |
| 47            | L walking                 | Saint-Gaudens          |
| 48            | Lhl classic, capped       | Reich                  |
| 49            | Lhr draped bust           | Scot                   |
| 50            | Lhl coronet               | Longacre               |

|                          |                                 |
|--------------------------|---------------------------------|
| LIBERTY                  | 1793                            |
| LIBERTY                  | 1794                            |
| LIBERTY                  | 1795                            |
| LIBERTY                  | ***** 1796                      |
| LIBERTY                  | ***** 1809                      |
| LIBERTY                  | ***** 1840                      |
| LIBERTY                  | 1793                            |
| LIBERTY                  | 1793                            |
| LIBERTY                  | 1793                            |
| LIBERTY                  | ***** 1816                      |
| UNITED STATES OF AMERICA | 1856                            |
| LIBERTY                  |                                 |
| UNITED STATES OF AMERICA | 1859                            |
| IN GOD WE TRUST          | LIBERTY 1909                    |
| IN GOD WE TRUST          | 1864                            |
| UNITED STATES OF AMERICA | 1851                            |
| LIBERTY                  |                                 |
| UNITED STATES OF AMERICA | 1865                            |
| LIBERTY                  | ***** 1883                      |
| LIBERTY                  | 1913                            |
| IN GOD WE TRUST          | LIBERTY 1938                    |
| LIBERTY                  | ***** 1794                      |
| LIBERTY                  | ***** 1837                      |
| UNITED STATES OF AMERICA | 1837                            |
| LIBERTY                  | ***** 1837                      |
| IN GOD WE TRUST          | 1892                            |
| LIBERTY                  | IN GOD WE TRUST 1716            |
| LIBERTY                  | IN GOD WE TRUST 1946            |
| LIBERTY                  | ***** 1875, (+) IN GOD WE TRUST |
| IN GOD WE TRUST          | ***** LIBERTY 1916              |
| IN GOD WE TRUST          | LIBERTY 1932                    |
| LIBERTY                  | ***** 1836                      |
| IN GOD WE TRUST          | LIBERTY 1916                    |
| LIBERTY                  | IN GOD WE TRUST 1948            |
| LIBERTY                  | IN GOD WE TRUST 1964            |
| LIBERTY                  | E • PLURIBUS • UNUM ***** 1878  |
| IN GOD WE TRUST          | LIBERTY 1921                    |
| IN GOD WE TRUST          | LIBERTY 1971                    |
| IN GOD WE TRUST          | LIBERTY 1979                    |
| LIBERTY                  | UNITED STATES OF AMERICA        |
| LIBERTY                  | ***** 1796                      |
| LIBERTY                  | ***** 1809                      |
| LIBERTY                  | ***** 1908                      |
| <del>LIBERTY</del>       | <del>***** 1907</del>           |
| LIBERTY                  | ***** 1907                      |
| LIBERTY                  | ***** 1913                      |
| LIBERTY                  | ***** 1796                      |
| LIBERTY                  | ***** (no date!)                |
| LIBERTY                  | ***** 1849                      |





cent ~~minted~~ 1855 Philadelphia ~~total~~ 6 1,574,229

~~1~~ @ \$80.00

obverse: condition: EF-40  
subject: Liberty head facing left with coronet  
inscription: "LIBERTY & JUSTICE 1851" ~~minted~~  
designer: Christian Gobrecht

2 4  
1 @ \$80.00

reverse: condition: EF-40  
subject: wreath  
inscription: "UNITED STATES OF AMERICA ONE CENT"  
designer: Christian Gobrecht

~~1~~ @ \$20.00  
~~1~~ @ \$50.00  
~~1~~ @ \$93.00

edge: [plain]

weight: 10.89 g <sup>3</sup> 2 (if not 2, then 1.000 is assumed)

composition: 1000 copper (unrecd)

diameter: 27.8 mm

notes: "1851" s/a "1851 [inverted]" on obverse.



1855 cent EF-40 1 @ \$80.00  
1795 half dollar G-4/AG-3 1 @ \$525.00  
1870cc dollar F-12 1 @ \$225.00

Color

~~data structures~~  
~~condition~~  
cond / year

condition

\* t t t s s s s

where

ttl = T  
ssss = S

~~SHIELDON:~~

|    |    |       |
|----|----|-------|
| S! | 0  | AG-03 |
|    | 1  | G-04  |
|    | 2  | VG-08 |
|    | 3  | F-12  |
|    | 4  | VF-20 |
|    | 5  | VF-30 |
|    | 6  | EF-40 |
|    | 7  | EF-45 |
|    | 8  | AU-50 |
|    | 9  | AU-55 |
|    | 10 | MS-60 |
|    | 11 | MS-63 |
|    | 12 | MS-65 |
|    | 13 | MS-67 |
|    | 14 | MS-70 |

offset  
in byte  
value  
table  
shieldon.

TEXTCOND:

|    |   |       |
|----|---|-------|
| T: | 0 | AG    |
|    | 1 | G     |
|    | 2 | VG    |
|    | 3 | F     |
|    | 4 | VF    |
|    | 5 | EF    |
|    | 6 | AU    |
|    | 7 | MS    |
|    | 8 | Proof |

offset  
in addr table  
of word →  
TEXTCOND.

weight

3 bytes (BCD)

min 00.0000  
max 99.9999

a a . b b c c (g)

diameter

2 bytes (BCD)

a a . b b (mm)

compos

- 0 copper
- 1 nickel
- 2 tin and zinc
- 3 zinc
- 4 silver
- 5 manganese
- 6 steel
- 7 gold

year (good → 2048)

offset from 1793

edges

0 blank  
1 wickered  
2 plain  
3 other

bars  
vine (slower or faster)  
griped  
receded

edges

edges 1 2 3

- 0 plain
- 1 wickered
- 2 lettered
- 3 other





~~United States Regular Issues~~

style

| type                    | value (\$) | years minted | # years | 71 sub | obverses                                      |
|-------------------------|------------|--------------|---------|--------|-----------------------------------------------|
| 0 half cent             | 0.005      | 1793 - 1857  | 65      |        | 0, 1, 2, 4 <sup>8</sup> , 4, 5                |
| 1 cent                  | 0.01       | 1793 - date  | 176 +   | *      | 6, 7, 8, 2, 4 <sup>8</sup> , 4, 9, 11, 12, 13 |
| 2 two-cent piece        | 0.02       | 1864 - 1873  | 10      |        | 14                                            |
| 3 three-cent piece (Ag) | 0.03       | 1851 - 1873  | 23      |        | 15                                            |
| 4 three-cent piece (Ni) | 0.03       | 1865 - 1889  | 25      |        | 16                                            |
| 5 five-cent piece       | 0.05       | 1866 - date  | 123 +   | X      | 14, 17, 18, 19                                |
| 6 half dime             | 0.05       | 1795 - 1873  | 79      |        | 20, 21, 23                                    |
| 7 dime                  | 0.10       | 1796 - date  | 173 +   | *      | 3, 42, 23, 24, 25, 26                         |
| 8 twenty-cent piece     | 0.20       | 1875 - 1878  | 4       |        | 27                                            |
| 9 quarter dollar        | 0.25       | 1796 - date  | 193 +   | X      | 3, 42, 23, 24, 29, 30                         |
| 10 half dollar          | 0.50       | 1794 - date  | 195 +   | *      | 20, 3, 42, 31, 23, 24, 32, 33, 34             |
| 11 dollar (Ag)          | 1.00       | 1794 - date  | 195 +   | X      | 20, 3, 23, 28, 35, 36, 37, 38                 |
| 12 dollar (Au)          | 1.00       | 1849 - 1889  | 41      | *      | 49, 39, 40                                    |
| 13 quarter eagle        | 2.50       | 1796 - 1929  | 134     | X      | 41, 42, 10, 22, 5, 43                         |
| 14 three-dollar piece   | 3.00       | 1854 - 1889  | 36      |        | 40                                            |
| <hr/>                   |            |              |         |        |                                               |
| 16 half eagle           | 5.00       | 1795 - 1729  | 135     | X      | 41, 42, 10, 22, 5, 43                         |
| 17 eagle                | 10.00      | 1795 - 1933  | 139     | X      | 41, 5, 46                                     |
| 18 double eagle         | 20.00      | 1849 - 1933  | 85      |        | 50, 47                                        |

gethex1

is bas, y, 2 (starting value)  
(scrupus)

o: asto  
dest: htab

1. print 2 in hexide

2. get value → asto

3. print asto normal

pitch width: 13.2"  
paper length: 25.4"

sec  
sbc #7

htab  
fix bit

edit:

~~A9A4:DD~~

~~A9A5:48~~

A908: cmp #58D  
jsr workfid

20 A9 A9

A915: jmp edrefet

A9B7  
1101  
monthtbl 13th month  
typetbl (tp)  
typetbl  
statbl

1256, 1295 p.to3

~~asto~~  
↓  
0001020304050607  
↑  
htab

~~240~~

Months

return  
displays  
wrong (240 = 0 into wfbait)  
(curaddr)

10 11 12 13 14 15 16 17  
00 01 02 03 04 05 06 07  
↑  
asto2  
x=0

|        | # bits | values | (range)             |
|--------|--------|--------|---------------------|
| Year   | 8      | 0-127  | 1900-2027           |
| Month  | 4      | 0-15   | (0)-12 (13, 14, 15) |
| Day    | 5      | 0-31   | 0-31                |
| Hour   | 8      | 0-255  | 0-23 (24-255)       |
| Minute | 8      | 0-255  | 0-59 (60-255)       |

↓  
0 1 2 3 4 5 6 7  
asto = 0 0 0 0 0 0 0 0 ← current value  
dnb 0 0 0 0 0 0 0 0  
asto2 = 0 0 0 1 0 0 0 0 ← which bit we're at  
asto3 = 0 0 1 0 0 0 0 0  
1101111

# fields and how they're displayed

| #  | abbr | name                | dtype | display                     | (file # → first column position) |
|----|------|---------------------|-------|-----------------------------|----------------------------------|
| 0  | st   | storage type        | 6     | \$x.. <stor>                |                                  |
| 1  | nl   | name length         | 7     | \$x                         | (high underlines below name)     |
| 2  | nam  | name                | 8     | cccccccccccc                |                                  |
| 3  | typ  | type                | 9     | \$xx.. <typ>                |                                  |
| 4  | kb   | key block           | 1     | \$xxxx                      |                                  |
| 5  | bu   | blocks used         | 1     | \$xxxx                      |                                  |
| 6  | len  | length              | 2     | \$xxxxxx                    |                                  |
| 7  | cre  | creation            | 5     | \$xxxx, \$xxxx.. <datetime> |                                  |
| 8  | ver  | version --> minimum | 4     | xx.. --> xx                 |                                  |
| 9  | acc  | access              | 10    | \$xx.. DMB \$\$\$WR         |                                  |
| A  | aux  | auxiliary           | 1     | \$xxxx                      |                                  |
| B  | mod  | last modification   | 5     | \$xxxx, \$xxxx.. <datetime> |                                  |
| C  | hdb  | header block        | 1     | \$xxxx                      |                                  |
| D  | res  | reserved            | 3     | xxxxxxxxxxxxxxxx            |                                  |
| E  | enl  | entry length        | 0     | \$xx                        |                                  |
| F  | epb  | entries per block   | 0     | \$xx                        |                                  |
| 10 | flc  | file count          | 1     | \$xxxx                      |                                  |
| 11 | bit  | bit map block       | 1     | \$xxxx                      |                                  |
| 12 | bpv  | blocks per volume   | 1     | \$xxxx                      |                                  |
| 13 | pbk  | parent block        | 1     | \$xxxx                      |                                  |
| 14 | pen  | parent entry number | 0     | \$xx                        |                                  |
| 15 | pel  | parent entry length | 0     | \$xx                        |                                  |

934 10016

dtype:

|   |    |                             |
|---|----|-----------------------------|
| 4 | 0  | \$xx                        |
| 8 | 1  | \$xxxx                      |
| 1 | 2  | \$xxxxxx                    |
| 1 | 3  | xxxxxxxxxxxxxxxx            |
| 1 | 4  | xx.. --> xx                 |
| 2 | 5  | \$xxxx, \$xxxx.. <datetime> |
| 1 | 6  | \$x.. <storage type>        |
| 1 | 7  | \$x (w/ high underlines)    |
| 1 | 8  | cccccccccccc                |
| 1 | 9  | \$xx.. <type>               |
| 1 | 10 | \$xx.. DMB \$\$\$WR         |

yes.



DATE

date:

|                     |                                                                                                                         |                |                           |                                                                                                                                        |
|---------------------|-------------------------------------------------------------------------------------------------------------------------|----------------|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
|                     | <u>A</u>                                                                                                                | <u>C</u>       | <u>date</u>               | <u>date</u>                                                                                                                            |
| lcr date            |                                                                                                                         | m <sub>0</sub> | $\phi y_1, y_2, y_3, y_4$ | m <sub>0</sub> m <sub>1</sub> m <sub>2</sub> m <sub>3</sub> d <sub>0</sub> d <sub>1</sub> d <sub>2</sub> d <sub>3</sub> d <sub>4</sub> |
| <del>ror date</del> |                                                                                                                         | d <sub>4</sub> |                           | <del>m<sub>0</sub>m<sub>1</sub>m<sub>2</sub>m<sub>3</sub>d<sub>0</sub>d<sub>1</sub>d<sub>2</sub>d<sub>3</sub>d<sub>4</sub></del>       |
| ldz date+1          | m <sub>1</sub> m <sub>2</sub> m <sub>3</sub> d <sub>0</sub> d <sub>1</sub> d <sub>2</sub> d <sub>3</sub> d <sub>4</sub> | m <sub>0</sub> | $\phi y_0$ — $y_6$        | m <sub>1</sub> — m <sub>3</sub> d <sub>0</sub> — d <sub>4</sub>                                                                        |
| ror                 | m <sub>0</sub> m <sub>1</sub> m <sub>2</sub> m <sub>3</sub> d <sub>0</sub> d <sub>1</sub> d <sub>2</sub> d <sub>3</sub> | d <sub>4</sub> |                           |                                                                                                                                        |

lcr  
lcr  
lcr  
~~lcr~~  
~~ldz date~~ — ~~ldz date~~  
ldz lcr  
ldz monthtbl, x  
clt add  
inx  
ldz monthtbl, x  
sta add+1  
~~ldz~~  
jcr pstr

~~ldz~~  
~~ldz~~  
~~ldz~~

ldz #AP  
jcr out  
ldz date+1  
and #1F

asl  
asl  
asl  
sta date+1  
sed

ldx #5  
lda #16  
sta asto  
asl date+1  
adc asto  
sta asto  
dex  
bpl ddp1  
cld

lda asto  
jcr out  
lda #16  
jcr out  
ldx #AP  
jcr out

dollr1

d0 d1 d2 d3 d4

d0 d1 d2 d3 d4

hex to dec:

14. qsto: to convert  
out addr: 10, hi decimal bytes

sed  
ldx #7  
lda #1  
sta addr  
sta addr+1

17

asl qsto  
lda addr  
adc addr  
sta addr  
lda addr+1  
adc addr+1  
sta addr+1

dex  
bpl

cld  
rts

September 10, 1966  
May 20, 1977

June 12, 1974  
July 1, 1980

hex to dec: ~~BA71~~ BA71  
dsp date: ~~BABD~~ BABD  
ctrlc: BB18

2 → 10

$$(d_0, d_1, d_2, d_3, d_4)_2 \quad \text{[scribble]} = (x)_{10}$$

$$\begin{aligned} \vec{r}_0 + \vec{r}_1 + d_0 &= \vec{r}_1 \\ \vec{r}_1 + \vec{r}_1 + d_1 &= \vec{r}_2 \end{aligned}$$

(seq)

grid      x      z

~~sta sta~~    ~~d<sub>0</sub>, d<sub>1</sub>, d<sub>2</sub>, d<sub>3</sub>~~    z

ida # φ                  oooooo    o

~~[scribbled]~~                  [scribbled] →

~~[scribbled]~~                  [scribbled] →

~~[scribbled]~~                  [scribbled] →

ask grid                  ooφφd<sub>2</sub>d<sub>2</sub>d<sub>3</sub> 7 φφφφφφφφ  
φφd<sub>2</sub>d<sub>2</sub>d<sub>2</sub>d<sub>2</sub>d<sub>2</sub>φ 7 φ ———— φ

| seed                          | $\frac{n}{a}$         | $\frac{a}{t}$                                    | $\frac{t}{r}$ |
|-------------------------------|-----------------------|--------------------------------------------------|---------------|
| sta $n$                       | (#) $a b c d e f g h$ | -                                                | -             |
| <del>sta <math>n</math></del> | $a b c d e f g h$     | $\phi$                                           | $\phi$        |
| <del>sta <math>n</math></del> | $a b c d e f g h$     | $\phi$                                           | $\phi$        |
| tdx $\neq 7$                  |                       |                                                  |               |
| ldz $\neq \phi$               |                       |                                                  |               |
| → ast $n$                     | b c d e f g h $\phi$  | $\phi \phi \phi \phi \phi \phi$                  | a $\phi$      |
| <del>ldz <math>n</math></del> |                       | $\phi$                                           |               |
| adc $t$                       |                       | $\phi \phi \phi \phi \phi$                       |               |
| <del>sta <math>n</math></del> |                       | <del><math>\phi \phi \phi \phi \phi</math></del> |               |
| dex                           |                       |                                                  |               |
| bpl                           |                       |                                                  |               |

setfpos

~~lda #5~~  
~~lda #5~~  
lda frc, x  
sta vtab  
jsr bascale  
lda fht, x  
~~lda #5~~  
~~lda #5~~  
rts

1: X:field #  
0: bas, ~~hlt~~ ~~(hlt)~~  
↑ ~~setfpos~~ ~~hlt~~  
~~hlt~~

lda #5  
jsr setfpos  
0: bas  
~~hlt~~  
~~x = (frc) E~~  
hlt

6x  
~~lda hlt~~  
~~lda hlt~~  
jsr hlt  
sta hlt  
lda #14  
lda (curaddr), y  
lda hlt  
jsr hlt  
lda hlt  
lda #13  
lda (curaddr), y  
lda hlt  
jsr hlt  
~~hlt~~

setfpos: i: X (field #)  
0: bas  
2 (hlt)

ldi buf,addr+1  
 rdc #2  
 stg buf,addr+1

4

2

4

10

\*

inc buf,addr+1  
 inc buf,addr+1

12

readr: ldi # < dirbuf 2  
 stg rdx+1 4  
 ldi # > dirbuf 2  
 stg rdx+2 4

[ MLI CALL ]

rdx: ldi # 5+2 2  
 rdx: ldi \$FFFF, x 4  
 stg block-2, x 5  
 ldx 2 2  
 spv # 4 2  
 bne rdx 2, 3 if br

2

4

2

4

2

4

5

2

2

3

4

5

2

2

45

45

readr: ldi # < dirbuf 2  
 stg rdx+1 4  
 stg rdx+1+1 4  
 ldi # > dirbuf 2  
 stg rdx+2 4  
 ldx rdx+2 4

[ MLI CALL ]

rdx: ldi # 5+2 2  
 rdx: ldi \$FFFF, x 4  
 stg block 4  
 ldx 2 2  
 ldi \$FFFF, x 4  
 stg block+1 4

2

4

4

2

4

4

2

4

4

2

4

4

40

40

42 45

readr: ldi # < dirbuf 2  
 stg addr 4  
 ldi # > dirbuf 2  
 rdx addr+1 4

[ MLI CALL ]

rdx: ldi # 5+2 2  
 ldi (addr), x 5  
 stg block 4  
 ldx 2 2  
 ldi (addr), x 5  
 stg block+1 4

2

4

2

4

2

5

4

2

5

4

10 145

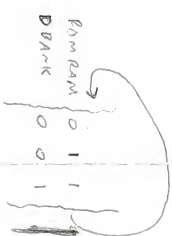
34

# COMMAND SYSTEM

MAXIMUM OF 832 COMMANDS IN COMMAND/COMMANDS

c 4P3AS ~~NA~~  
= F2-FD

BRANK



0837

ADC # 57F

$$\sqrt{13}$$

南塘

50

60

70

80

0.6

74

00001110

227

22

8111

1

227

$$M_{\text{max}} = 8\phi$$
[illegible]

① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨ ⑩ ⑪ ⑫ ⑬ ⑭ ⑮ ⑯ ⑰ ⑱ ⑲ ⑳ ㉑ ㉒ ㉓ ㉔ ㉕ ㉖ ㉗ ㉘ ㉙ ㉚ ㉛ ㉜ ㉝ ㉞ ㉟ ㊱ ㊲ ㊳ ㊴ ㊵ ㊶ ㊷ ㊸ ㊹ ㊺ ㊻ ㊼ ㊽ ㊾ ㊿

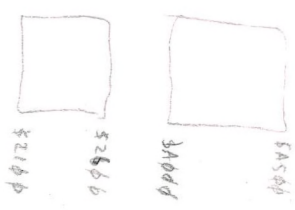
$$A = \phi 0000 f_h$$

A = bbbbbb + 0000101



COMMAND SYSTEM

MAC      \$2100      LENGTH \$45  
DEST      \$A000



Part 1

~~MEADHAM, George A. 11/10~~

JAN 20 1964

PREFIX: 7  
 10. 1. 2. 3. 4. 5. 6. 7.  
 1. M I 1 Q R S 7.

~~A = 1~~  
~~A = 2~~  
~~A = 3~~

[illegible]

60

output

~~1~~

|    |      |    |      |
|----|------|----|------|
| 10 | 4.75 | 10 | 4.75 |
| 10 | 4.75 | 10 | 4.75 |

1774

$\vdots$

0555 1111

driven (1, 2)

Plot (0-7)

$$1111 = 1496h(1-f)$$

delever:

19  
20  
21  
22  
23

124,127

518,527

133

376,372

2523

9119

2558-20

MA-1010  
28E4

up to  
30 file names on screen

111010101011101010

2250

2270 < 2250.244F

267,278

2050

9212

NAMELIST:

ABORT

(1972) 41x3 db

9130  
91C5

where T = type (81)

242

235

251

260

264

267

270

273

276

279

282

285

288

291

294

297

300

303

306

309

312

315

318

321

243

252

255

258

261

264

267

270

273

276

279

282

285

288

291

294

297

300

303

306

309

312

315

1000

1001

1002

1003

1004

1005

1006

1007

1008

1009

1010

1011

1012

1013

1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013

STACK

400E

400E

400E

400E

400E

400E

400E

400E

400E

400E

400E

400E

NAMELIST: 2381  
FILIC: 82  
EXTREME: 83  
EXTREME: 84

NAMELIST: 2381  
FILIC: 82  
EXTREME: 83  
EXTREME: 84

NAMELIST: 2381  
FILIC: 82  
EXTREME: 83  
EXTREME: 84

next byte is code

front end loader

new command

~~front end loader~~

7FD: SMP 34B97

828: 33 b1 b8 1c 4b 8b 4b ; 34428 (in pairs)

1b 3b 4b 1b 4b  
(27 total)

865: LDA 34428, Y ; Y = 4 initially

Becq 34B48

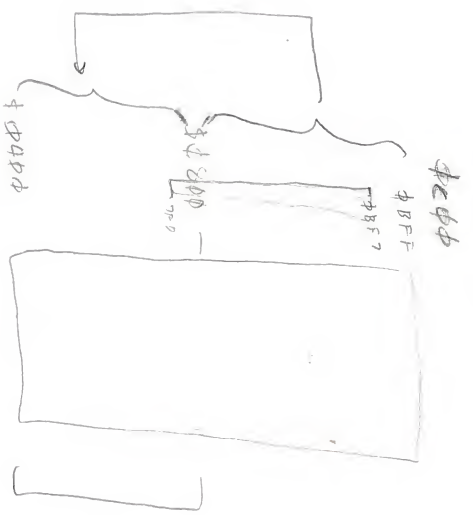
37A 3FC

37A 3EF

37Y 3F1

163

BAB:



Z-P G

text placed

EF: 41

FD: 14 (offset of first 1/2 pair is 34B48) used as variable

F1: 34B48 Y 510

F7: 34B48 first track

FB: 34

FC: 43

FD: 41

FE: 12

